PassQueen
a.k.a dumpsqueen.com

HCIA-AI V3.0

Huawei H13-311_V3.0

Version Demo

Total Demo Questions: 38

Total Premium Questions: 389

Buy Premium PDF

https://passqueen.com

support@passqueen.com

Topic Break Down

Topic	No. of Questions
Topic 1, Artificial Intelligence Overview	45
Topic 2, Machine Learning Overview	86
Topic 3, Deep Learning Overview	76
Topic 4, Mainstream AI Development Frameworks	59
Topic 5, MindSpore Overview	13
Topic 6, Huawei AI Computing Platform	34
Topic 7, Huawei Cloud AI Development Platform	6
Topic 8, Huawei Cloud AI Services	36
Topic 9, Mix Questions	34
Total	389

QUESTION NO: 1

Huawei's chip support HUAWEI HiAI Which module of?

A. HiAI Engine

B. HiAI Foundation

C. HiAI Framework

D. HiAI Service

ANSWER: B

Explanation:

HiAI Foundation is correct because it is the foundational, chip-facing part of the HUAWEI HiAI platform. This layer exposes and coordinates the underlying AI-computing capabilities provided by Huawei hardware, including heterogeneous processing resources such as the CPU, GPU, and dedicated neural-processing hardware. It provides the basic capabilities on which the higher-level HiAI software stack can build AI functions, allowing applications and frameworks to use acceleration supplied by the device chipset rather than relying only on general-purpose computation.

Huawei introduced HiAI as a mobile AI computing platform designed to make the AI capabilities of its Kirin chipsets available to developers and applications. In that architecture, the foundation layer represents the hardware-capability base: it is the part most directly associated with support from Huawei chips. This is consistent with Huawei's description of HiAI as a platform that leverages on-device AI processing and dedicated AI hardware. See Huawei's HiAI platform announcement and its overview of the mobile AI ecosystem.

QUESTION NO: 2

As the following, what are the Python language design philosophy? (Multiple Choice)

A. Beautiful

B. Expensive

C. Explicit

D. Simple

ANSWER: A C D

Explanation:

Python's design philosophy is commonly summarized by "The Zen of Python," a collection of aphorisms authored by Tim Peters and published as PEP 20. It begins with "Beautiful is better than ugly," reflecting Python's emphasis on readable, clean, and maintainable code. In practice, this encourages programmers to write code whose intent is easy for other people to understand, review, and modify.

"Explicit is better than implicit" is another central principle. Python favors code that clearly states what it does instead of relying on hidden behavior, unclear conventions, or surprising side effects. This supports reliability and makes programs easier to debug.

"Simple is better than complex" also directly appears in the Zen. Python aims to provide expressive language features while encouraging straightforward solutions where possible. Simplicity helps reduce unnecessary implementation complexity and improves long-term maintainability. These principles are foundational guidance for Python programming style and language design, rather than merely optional preferences. The official wording is available in PEP 20: The Zen of Python; it can also be displayed in a Python interpreter by running `import this`. Python's official documentation also presents the language's emphasis on readability in its Tutorial.

QUESTION NO: 3

The pooling layer in the convolutional neural network can reduce the size of the lower layer input. Common pooling is:

A. Minimum strata

B. Product pooling layer

C. Maximum pooling layer

D. Average pooling layer

ANSWER: C D

Explanation:

Maximum pooling layer and Average pooling layer are the standard, commonly used pooling operations in convolutional neural networks. Pooling applies a sliding window over local regions of an input feature map and replaces each region with one representative value. This reduces the spatial height and width of the feature maps, lowering subsequent computation and memory requirements while retaining useful local information.

A Maximum pooling layer selects the largest activation in each pooling window. It is particularly useful for preserving the strongest detected feature response in a local area, such as an edge, texture, or learned visual pattern. An Average pooling layer calculates the mean value of the activations in each window. This produces a smoothed summary of local feature responses and is also widely used, including global average pooling near the end of many CNN architectures.

Both operations are supported as fundamental CNN building blocks in mainstream deep-learning frameworks. TensorFlow describes max pooling and average pooling APIs at TensorFlow Max Pool and TensorFlow Average Pool. Their role in reducing feature-map dimensions aligns with the CNN knowledge assessed in HCIA-AI.

QUESTION NO: 4

The commonly used functions for mathematical operations in Python are basically in lhe math module and the cmath module.

A. True

B. False

ANSWER: A

Explanation:

True is correct. Python provides the `math` module as its standard library module for common mathematical functions involving real numbers. It includes functions and constants used frequently in numerical work, such as square roots, powers, logarithms, trigonometric functions, factorials, ceiling and floor operations, and constants including `pi` and `e`. The Python standard library also provides the `cmath` module for corresponding mathematical operations on complex numbers. Its functions are designed to accept and return complex values where appropriate, supporting operations such as complex square roots, logarithms, exponentials, and trigonometric calculations.

Therefore, these two modules are the principal standard-library sources of reusable mathematical functions: `math` focuses on real-number mathematics, while `cmath` supports complex-number mathematics. Python's official documentation describes `math` as providing access to mathematical functions defined by the C standard, and `cmath` as providing mathematical functions for complex numbers. See the official Python math module documentation and Python cmath module documentation.

QUESTION NO: 5

HUAWEI CLOUD EI Enable more use of corporate boundaries AI And big data services to accelerate business development and benefit society. HUAWEI CLOUD EI The service can serve the enterprise in the following

aspects?

A. Industry data

B. Industry wisdom

C. algorithm

D. Computing power

ANSWER: A B C D

Explanation:

Huawei Cloud Enterprise Intelligence (EI) is designed as an enterprise AI and big-data capability set, rather than as a single isolated application. It helps organizations make their data usable through cloud data services, then applies AI capabilities and industry-oriented intelligence to turn that data into operational insights and business value. Therefore, **Industry data** is a core service aspect because enterprise AI solutions require data ingestion, storage, governance, analysis, and use of domain data. **Industry wisdom** is also included because EI targets practical intelligence for business scenarios and industry needs, allowing organizations to convert accumulated experience and knowledge into intelligent services. **algorithm** is fundamental: machine-learning and AI algorithms provide the models and analytical methods that identify patterns, make predictions, and support automated decisions. Finally, **Computing power** supplies the scalable cloud infrastructure needed to train models, process large datasets, and run AI workloads efficiently. Together, data, industry intelligence, algorithms, and compute form the essential foundation for enterprise AI adoption and align with Huawei Cloud EI's purpose of making AI and big-data capabilities available to enterprises. Huawei Cloud's EI overview is available at Huawei Cloud EI; its broader AI services portfolio is described at Huawei Cloud AI.

QUESTION NO: 6

The following about the general form identification service returned type The field statement is correct?

A. type Representative text recognition area type

B. type for text Time represents the text recognition area

C. type Representative form type

D. type for table Time represents the form recognition area

ANSWER: B D

Explanation:

In the General Table Recognition response, `type` classifies each detected recognition region. A region whose `type` value is `text` is a text recognition area. This identifies content that is recognized as ordinary text rather than as a structured table. A region whose `type` value is `table` is a table recognition area. This identifies content that the service has detected and processed as tabular structure, allowing the response to distinguish table content from surrounding textual content. Therefore, the statements that explicitly map `text` to a text recognition area and `table` to a form/table recognition area accurately describe the possible meanings of this returned field. This classification is useful when an application needs to route textual regions and tabular regions to different downstream parsing or storage logic. Huawei Cloud documents OCR service APIs and their response fields in its OCR documentation; the corresponding service operations can also be reviewed through the official Huawei Cloud API Explorer.

QUESTION NO: 7

What model is not a cyclic neural network?

A. RNN

B. LSTM

C. GBDT

D. GRU

ANSWER: C

Explanation:

GBDT is the correct answer. The abbreviation is conventionally written as GBDT, meaning Gradient Boosting Decision Tree. It is an ensemble-learning method that builds a sequence of decision trees, with each successive tree trained to reduce residual errors from the preceding ensemble. Although this training process is iterative, it does not make GBDT a recurrent (cyclic) neural network: it has no recurrent hidden state, no neural-network sequence cells, and no feedback connections for retaining information across time steps. GBDT is commonly used for structured or tabular data and combines many weak decision-tree learners through gradient boosting. Scikit-learn's documentation describes gradient-boosted trees as an additive model constructed in a forward stage-wise fashion: scikit-learn Gradient Boosting documentation. In contrast, a recurrent neural network is designed to process sequential inputs by maintaining state across sequence positions; this sequence-processing architecture is described in the TensorFlow guide to recurrent neural networks. Therefore, GBDT belongs to tree-based machine learning rather than the recurrent-neural-network family.

QUESTION NO: 8

What are the common types of dirty data?

A. Malformed value

B. Duplicate value

C. Logically wrong value

D. Missing value

ANSWER: A B C D

Explanation:

Dirty data is data that does not reliably represent the intended real-world fact and therefore needs to be identified and treated during data preprocessing. **Malformed value** is a common category because data can violate its expected representation or format, such as an invalid date, malformed identifier, or a number stored with impermissible characters. **Duplicate value** is also common when the same entity or event is recorded more than once, which can distort counts, distributions, and model training results. **Logically wrong value** describes data that may have a valid format but is inconsistent with business rules or real-world logic, such as an impossible age or an end time preceding a start time. **Missing value** is another standard data-quality issue, occurring when an expected attribute has no recorded value. In AI workflows, these issues are typically assessed before training so that cleaning, deduplication, validation, and missing-value handling can be applied consistently. Huawei ModelArts documentation places data preparation and management within the machine-learning workflow; see Huawei Cloud ModelArts Documentation. For a broader description of data-quality dimensions and cleansing practices, see IBM: What is data quality?.

QUESTION NO: 9

Which of the following are advantages of using a knowledge graph in an intelligent application? (Multiple Choice)

A. Relationship query

B. Knowledge reasoning

C. Integration of related heterogeneous data

D. Elimination of all data-quality requirements

ANSWER: A B C

Explanation:

A knowledge graph represents entities and their relationships in a structured graph form. It supports relationship queries, such as finding the diseases associated with a symptom or the products associated with a customer. Its explicit entity-relation structure can support knowledge reasoning and provide interpretable relationship paths for intelligent applications. It can also integrate knowledge from heterogeneous sources after entities and relationships are modeled. A knowledge graph does not eliminate the need for data collection and governance; high-quality source data remains essential. Huawei Cloud Graph Engine Service provides graph data capabilities for relationship-oriented data scenarios. See Huawei Cloud Graph Engine Service.

QUESTION NO: 10

TensorFlow2.0 The methods that can be used for tensor merging are?

A. join

B. concat

C. split

D. unstack

ANSWER: B

Explanation:

The correct answer is `concat`. In TensorFlow 2.0, `tf.concat()` merges two or more tensors along an existing dimension (axis). The input tensors must have compatible shapes: their dimensions must be equal except for the dimension specified by `axis`. For example, tensors shaped `(2, 3)` and `(2, 4)` can be concatenated along axis 1 to create a tensor shaped `(2, 7)`. This operation is commonly used when combining batches, feature vectors, or intermediate model outputs while retaining the rank of the original tensors. TensorFlow's API documentation defines `tf.concat(values, axis)` specifically as concatenating a list of tensors along one dimension. The TensorFlow guide on tensor operations also presents concatenation as the standard mechanism for joining tensor values along an axis. See the official tf.concat API documentation and the TensorFlow tensor guide for syntax, shape requirements, and examples.

QUESTION NO: 11

Which descriptions are correct about python's index? (Multiple Choice)

A. Index from left to right defaults from 0

B. Index from left to right defaults from 1

C. Index from right to left defaults from -1

D. Index from right to left defaults from 0

ANSWER: A C

Explanation:

Python sequences, including strings, lists, and tuples, support positional indexing in both directions. The description "Index from left to right defaults from 0" is correct because the first item in a sequence has index `0`, followed by `1`, `2`, and so on. For example, in `items = ['a', 'b', 'c']`, `items[0]` returns `'a'`. Python also supports negative indexes, so the description "Index from right to left defaults from -1" is correct. Negative indexing starts at the end of the sequence: `items[-1]` returns the final item, `'c'`; `items[-2]` returns the item before it. This indexing model is useful when code needs to access the first or last elements without calculating sequence length. Python's official sequence-type documentation defines both zero-based positive indexing and negative indexing, while the language reference describes subscription using integer index values. See the Python common sequence operations documentation and the Python language reference on subscriptions.

QUESTION NO: 12

Huawei Cloud EI builds enterprise intelligence services based on three-tier services.

Which of the following options does not belong to Layer 3 services?

A. Basic platform services

B. General domain services

C. Industry sector services

D. Integration services

ANSWER: D

Explanation:

Integration services is the correct choice because it is not one of the three service layers used to describe Huawei Cloud Enterprise Intelligence (EI). Huawei Cloud EI organizes its enterprise intelligence capabilities as a layered service system comprising basic platform services, general domain services, and industry sector services. This model starts with underlying AI and data capabilities, builds reusable capabilities for common enterprise scenarios, and then provides solutions tailored to particular industries. “Integration services” may describe work performed when connecting systems, data sources, or applications, but it is not the name of a defined tier in this three-layer EI service classification. Therefore, when the question asks for the item that does not belong to Layer 3 services, integration services is the appropriate selection because it is outside the stated EI tier taxonomy rather than a recognized service layer within it. Huawei Cloud describes EI as a platform for enterprise intelligence and AI capabilities in its Enterprise Intelligence product information; related EI documentation is available in the Huawei Cloud EI Product Description.

QUESTION NO: 13

During the two classification process, we can set any category as a positive example.

A. TRUE

B. FALSE

ANSWER: A

Explanation:

TRUE is correct. In binary classification, the terms “positive” and “negative” designate the two class labels used to frame a prediction task; they do not inherently indicate that one category is better, more important, or more frequent. Therefore, either of the two original categories may be selected as the positive class, provided that the choice is applied consistently throughout dataset labeling, model training, inference output interpretation, and evaluation. For example, a classifier distinguishing “approved” from “rejected” applications may use either outcome as the positive label depending on which event the business wants to detect or measure. This selection affects the interpretation of metrics such as precision, recall, true-positive rate, and false-positive rate, because those measures are calculated with respect to the selected positive class. Huawei ModelArts supports supervised-learning workflows in which labels are defined for the task dataset and then used consistently by training and evaluation processes; see the Huawei Cloud ModelArts User Guide. The same positive-label convention underlies binary-classification evaluation, including ROC-based measures, as described in the scikit-learn ROC metrics documentation.

QUESTION NO: 14

There are many commercial applications for machine learning services.

What are the main business scenarios covered? (Multiple Choice)

A. Financial product recommendation

B. Predictive maintenance

C. Telecom customer retention

D. Retailer grouping

ANSWER: A B C D

Explanation:

Machine learning creates business value by learning patterns from historical operational, transactional, customer, and device data, then producing predictions, rankings, or groups that support business decisions. Financial product recommendation is a core recommendation scenario: models use customer attributes and behavioral data to rank products likely to be relevant to each customer. Predictive maintenance applies classification, regression, and anomaly-detection techniques to equipment telemetry, enabling organizations to estimate failures or maintenance needs before an outage occurs. Telecom customer retention is a customer-churn scenario in which models identify customers with a higher likelihood of leaving, allowing targeted retention activity. Retailer grouping is a clustering and segmentation scenario, where retailers with similar characteristics or behavior are grouped to support differentiated operations, marketing, and resource allocation. These scenarios represent common practical uses of supervised learning, unsupervised learning, and recommendation capabilities in commercial settings. Huawei Cloud ModelArts provides AI development and deployment capabilities intended for industry applications such as intelligent recommendation, predictive analysis, and data-driven decision-making. See the Huawei Cloud ModelArts product page and ModelArts product documentation.

QUESTION NO: 15

GPU Good at computationally intensive and easy to parallel programs.

A. TRUE

B. FALSE

ANSWER: A

Explanation:

TRUE is correct. GPUs are designed for high-throughput computation: they contain many processing cores that can execute a large number of similar operations concurrently. This architecture is especially effective when a workload is computationally intensive and can be divided into many independent or similarly structured tasks, commonly described as data parallelism. Typical artificial-intelligence workloads have these characteristics. For example, tensor operations used in neural-network training and inference involve repeated matrix multiplications and element-wise calculations across large arrays of data. These operations can be distributed across many GPU cores, allowing substantially more work to be performed in parallel than on a processor optimized primarily for low-latency sequential execution.

The statement's reference to programs that are easy to parallelize is important: a GPU delivers its strongest performance gains when the same instruction pattern can be applied to many data elements with limited dependencies between tasks. Huawei's AI computing materials describe GPUs as heterogeneous-computing accelerators suited to parallel processing, while CUDA documentation explains the GPU programming model around executing many threads concurrently. See Huawei Ascend and NVIDIA CUDA C++ Programming Guide.

QUESTION NO: 16

What are the conditions for m row n column matrix A and p row q column matrix B to be multiplied?

A. m=p. n=q

B. n=p

C. m=n

D. p=q

ANSWER: B

Explanation:

The condition is **n=p**. A matrix with *m* rows and *n* columns has dimensions *m* × *n*, while a matrix with *p* rows and *q* columns has dimensions *p* × *q*. The product *AB* is defined only when the number of columns in the first matrix equals the number of rows in the second matrix. Therefore, the required compatible inner dimensions are *n* and *p*, so they must be equal.

When this condition holds, each entry of the result is calculated from a row of matrix A and a column of matrix B: corresponding elements are multiplied and then summed. Since A has *m* rows and B has *q* columns, the resulting product matrix has dimensions *m* × *q*. This dimensional rule is fundamental to linear algebra and is directly applicable to AI workloads, where tensors and matrices must have compatible shapes for operations such as linear transformations in neural-network layers. See the Wolfram MathWorld matrix multiplication reference and the NumPy matmul documentation for the corresponding matrix-dimension rule.

QUESTION NO: 17

Which of the following statements about universal form recognition services are correct?

A. rows Represents the line information occupied by the text block, the number is from 0 Start, list form

B. colums Represents the column information occupied by the text block, the number is from 0 Start, list form

C. The incoming image data needs to go through base64 coding

D. words Representative text block recognition result

ANSWER: A B C D

Explanation:

Universal form recognition returns both recognized text and layout-position metadata for each recognized text block. The `words` field is the text-recognition result for a block, allowing an application to obtain the actual characters identified by OCR. The `rows` field describes the row positions occupied by that text block, while `colums` denotes its column positions; these position values are supplied as zero-based lists. This structural metadata is important when extracting forms because it enables clients to reconstruct content according to its location in the original document rather than treating the OCR output as one unstructured text string. The request image is supplied as Base64-encoded image data, which lets binary image content be carried safely in the JSON request body. A client therefore reads the image bytes, Base64-encodes them, and places the resulting value in the applicable request field before calling the service. These response and request conventions support reliable integration with document-processing workflows. Huawei Cloud's OCR documentation provides the product overview and API usage model, including request image handling and recognition-result fields; see the Huawei Cloud OCR documentation and the Huawei Cloud OCR API Explorer.

QUESTION NO: 18

In deep learning tasks, when encountering data imbalance problems, which of the following methods can we use to solve the problem?

A. batch deletion

B. Random oversampling

C. Synthetic sampling

D. Random undersampling

ANSWER: B C D

Explanation:

Random oversampling, Synthetic sampling, and Random undersampling are established techniques for reducing the bias that can arise when one class has far fewer training examples than another. Random oversampling increases the representation of minority-class examples in the training set, allowing the model to encounter that class more frequently during optimization. Synthetic sampling creates new minority-class samples from relationships among existing minority examples; methods such as SMOTE are commonly used to improve minority coverage without merely repeating the same records. Random undersampling reduces the number of majority-class examples, producing a more balanced training distribution and potentially lowering training cost when the majority class is very large. These approaches are normally applied only to training data, while validation and test data should retain their natural class distributions so that evaluation reflects deployment conditions. In deep-learning workflows, resampling can also be combined with suitable metrics such as recall, precision, and F1 score to assess whether minority-class learning has improved. Scikit-learn documents random resampling utilities at sklearn.utils.resample, and imbalanced-learn documents synthetic minority oversampling at SMOTE.

QUESTION NO: 19

What are the scenarios or industries that are suitable for using Python? (Multiple Choice)

A. Artificial intelligence

B. Web development

C. Game development

ANSWER: A B C

Explanation:

Python is broadly applicable because its readable syntax, extensive standard library, and mature third-party ecosystem support rapid development across many domains. Artificial intelligence is a major Python use case: tools such as NumPy provide efficient numerical-array operations, while frameworks and libraries in the ecosystem support data preparation, model training, and inference. Web development is also well suited to Python, with widely used web frameworks enabling developers to build server-side applications, APIs, and services efficiently. Python is used in game development for gameplay scripting, tooling, automation, and prototyping, where fast iteration is especially valuable. Hardware development is another suitable scenario, particularly for hardware control, embedded prototyping, device testing, and automation. Python implementations and platforms such as MicroPython make Python available on constrained microcontrollers, while standard Python is frequently used to communicate with, test, and orchestrate connected devices. These uses reflect Python's role as a versatile general-purpose language rather than one limited to a single industry. See the official Python applications overview and the MicroPython project for examples of Python's broad application scope, including embedded and hardware-oriented development.

QUESTION NO: 20

What are the application scenarios for the break statement in the Python language? (Multiple Choice)

A. Any Python statement

B. while loop statement

C. for loop statement

D. Nested loop statement

ANSWER: B C D

Explanation:

In Python, `break` is used within a `while` loop statement to immediately terminate that loop when a required condition occurs, rather than waiting for the loop condition to become false. It is likewise valid in a `for` loop statement, where it stops iteration over the current iterable as soon as the desired item or stopping condition is reached. These uses are important for tasks such as searching for a matching record, ending input processing after a sentinel value, or stopping retries after success. A nested loop statement is also a valid application scenario. When `break` executes inside nested loops, it

terminates the innermost loop that lexically contains it; the enclosing loop can then continue unless the program uses additional control logic to stop it as well. This precise innermost-loop behavior enables targeted exits during multidimensional searches and similar processing. The Python language reference specifies that `break` may occur only syntactically nested in a `for` or `while` loop and terminates the nearest enclosing loop. See the Python language reference for break and the Python tutorial's loop-control guidance.

QUESTION NO: 21

HUAWEI CLOUD ModelArts Is for AI Which of the following functions are in the developed one-stop development platform Mode1Arts Can have?

A. Data governance

B. AI market

C. Visual workflow

D. Automatic learning

ANSWER: A B C D

Explanation:

Huawei Cloud ModelArts is designed as a one-stop AI development platform covering the end-to-end machine-learning lifecycle, so all listed capabilities belong to its functional scope. **Data governance** is supported through ModelArts data-management capabilities, which enable users to prepare, label, manage, and use datasets for training. **AI market** corresponds to the platform's AI asset-sharing capabilities, commonly presented through AI Gallery/marketplace-style resources where developers can discover and reuse algorithms, models, datasets, and AI applications. **Visual workflow** is provided through graphical workflow development, allowing users to compose AI processing and training steps visually rather than implementing every orchestration step manually. **Automatic learning** is available through ModelArts ExeML, Huawei Cloud's automated machine-learning capability, which helps users build and train models with reduced coding and algorithm-selection effort. Together, these functions support the intended ModelArts experience: organize data, obtain reusable AI assets, build workflows, automate model development, train models, and deploy AI services. Huawei's ModelArts overview describes the platform as covering AI development processes and related development tools, while its ExeML documentation specifically describes automated model development. See the Huawei Cloud ModelArts product description and ModelArts ExeML overview.

QUESTION NO: 22

The Python dictionary is identified by "{}", and the internal data consists of the key and its corresponding value.

A. True

B. False

ANSWER: A

Explanation:

True is correct. In Python, a dictionary is a mutable mapping type that stores associations between keys and values. Dictionary displays and dictionary literals use curly braces, with entries conventionally written as `key: value`, for example `{"model": "Ascend", "count": 2}`. Each key identifies the value associated with it, allowing a program to retrieve data efficiently by using the key, such as `device["model"]`. Keys in a dictionary must be hashable, and a given key maps to one current value within that dictionary; values may be of any Python object type. An empty dictionary is written as `{}`, while a populated dictionary retains the same brace-based syntax and separates key-value pairs with commas. This key-to-value mapping model is the defining behavior of Python dictionaries and is widely used to represent structured attributes, configuration data, and lookup tables in AI and general Python programs. Python's language reference formally defines dictionaries as mappings indexed by keys, and its standard-library documentation describes the `dict` type and its operations. See the Python Tutorial: Dictionaries and the Python Standard Types: dict.

QUESTION NO: 23

The main computing resources included in the Da Vinci architecture computing unit are?

A. Vector calculation unit

B. Scalar Computing Unit

C. Tensor computing unit

D. Matrix calculation unit

ANSWER: A B D

Explanation:

The Da Vinci architecture's AI Core organizes its primary execution resources around scalar, vector, and matrix-oriented computation. The scalar computing unit performs control-flow-related work and scalar operations, supporting such tasks as address calculation, loop control, comparisons, and instruction coordination. The vector calculation unit is designed for high-throughput element-wise processing, including common neural-network operations such as activation functions, normalization-related calculations, and vector arithmetic. The matrix calculation unit, also commonly called the Cube unit in Ascend documentation, accelerates dense matrix multiplication and convolution-style calculations that dominate much deep-learning workload execution. Together, these resources allow a model operator to combine control processing, element-level computation, and large-scale matrix operations efficiently within an AI Core. Huawei's Ascend C documentation describes the scalar, vector, and Cube computational resources and their roles in programming Ascend AI processors. See the Huawei Ascend C API Reference and the Huawei Ascend product documentation portal for the architecture and programming context.

QUESTION NO: 24

Which of the following about the dictionary in Python is correct? (Multiple Choice)

A. Each key and its corresponding value are separated by a colon (:).

B. Different key-value pairs are separated by commas (,).

C. The entire dictionary is enclosed in curly braces ({ }).

D. The keys of the dictionary are unique and the data type is uniform.

ANSWER: A B C

Explanation:

Python dictionary literals use a clear delimiter structure: a colon separates each key from the value associated with that key, as in `{"model": "Ascend"}`. When a dictionary contains more than one item, commas separate the individual key-value pairs, for example `{"model": "Ascend", "count": 2}`. The complete literal is enclosed in curly braces, `{}`; an empty pair of braces creates an empty dictionary. These rules make dictionary syntax easy to read and allow Python to parse the mapping correctly.

A dictionary maps unique, hashable keys to values. Values are not required to share one data type, so a single dictionary can validly contain text, integers, lists, and other objects as values. Python's language reference defines dictionary displays and their colon, comma, and brace syntax, while the standard library documentation describes dictionaries as mappings indexed by keys. See the Python language reference on dictionary displays and the Python documentation for dict mapping types.

QUESTION NO: 25

enter 32*32 Image with size 5*5 The step size of the convolution kernel is 1 Convolution calculation, output image Size is:

A. 28*23

B. 28*28

C. 29*29

D. 23*23

ANSWER: B

Explanation:

28*28 is correct. With no padding specified, convolution is treated as valid convolution: the 5×5 kernel can be placed only where its full area remains inside the 32×32 input image. For each spatial dimension, the convolution output-size formula is ⌊(input size + 2 × padding − kernel size) / stride⌋ + 1. Substituting an input size of 32, padding of 0, kernel size of 5, and stride of 1 gives ⌊(32 + 0 − 5) / 1⌋ + 1 = 28. The same calculation applies independently to both height and width because the input and kernel are square and the stride is identical in both directions. Therefore, there are 28 valid horizontal kernel positions and 28 valid vertical positions, producing a 28×28 feature map. This is the standard spatial-dimension calculation used by convolution operators; padding must be explicitly supplied to preserve a larger output size. See the convolution output-shape definition in PyTorch Conv2d documentation and the TensorFlow tf.nn.conv2d documentation.

QUESTION NO: 26

Python script execution mode includes interactive mode and script mode

A. True

B. False

ANSWER: A

Explanation:

True is correct. Python is commonly used in two primary execution modes: interactive mode and script mode. In interactive mode, a user starts the Python interpreter and enters statements directly at a prompt, receiving results immediately after each statement or expression is evaluated. This is particularly useful for learning the language, testing small pieces of code, and quickly inspecting objects or calculations. In script mode, Python statements are saved in a source file, conventionally with the `.py` extension, and the interpreter executes that file as a program, such as with `python my_program.py`. Script mode is the standard approach for creating reusable applications, automation tasks, and larger AI development projects. The official Python tutorial explicitly distinguishes use of the interpreter in interactive mode from executing Python source files as scripts. See the Python documentation on using the Python interpreter and the Python command-line documentation for the supported invocation and execution behavior.

QUESTION NO: 27

Which of the following are the cloud services provided by Huawei Cloud EI Visual Cognition? (Multiple choice)

A. Text recognition

B. Face recognition

C. Image recognition

D. Content detection

E. Image processing

ANSWER: A B C D E

Explanation:

Huawei Cloud EI Visual Cognition provides AI capabilities that analyze, understand, and transform visual information. Its service portfolio includes text recognition, commonly delivered through Optical Character Recognition capabilities that extract printed or handwritten text and structured information from images and documents. It also includes face recognition for detecting, comparing, and analyzing facial attributes in supported scenarios, as well as image recognition for identifying objects, scenes, labels, and other image content.

Content detection is part of visual-intelligence offerings because it analyzes visual material for specified content and supports content-governance and moderation use cases. Image processing is also a visual cognition capability: it applies intelligent enhancement, editing, restoration, style, or related transformations to images. Together, these services cover the principal stages of visual AI workloads: obtaining text and semantic meaning from images, recognizing people and objects, detecting relevant content, and processing images for downstream use.

Huawei Cloud describes its image-recognition capabilities and related visual AI scenarios on its Image Recognition service page. Its Optical Character Recognition service page documents the text-extraction capability used for text recognition workloads.

QUESTION NO: 28

In the classic convolutional neural network model,Softmax What hidden layer does the function follow?

A. Convolutional layer

B. Pooling layer

C. Fully connected layer

D. All of the above

ANSWER: C

Explanation:

In a classic convolutional neural network used for classification, the Softmax function typically follows the **Fully connected layer**. Convolutional layers learn local visual features, such as edges, textures, and increasingly complex patterns, while pooling layers reduce spatial dimensions and retain important feature responses. After these feature-extraction stages, the learned feature representation is commonly flattened or otherwise aggregated and passed to a fully connected layer. This layer produces one numerical score, often called a logit, for each possible class.

Softmax is then applied to this vector of class scores to convert it into a probability distribution. Each resulting value is between zero and one, and all output probabilities sum to one. This makes the model's output directly suitable for selecting the predicted class and for training with multiclass cross-entropy loss. Therefore, in the conventional CNN classification architecture described by the question, Softmax is positioned after the fully connected classification layer. TensorFlow's documentation describes Softmax as converting logits into a probability distribution: TensorFlow Softmax activation. The corresponding dense layer is documented as the standard fully connected neural-network layer: TensorFlow Dense layer.

QUESTION NO: 29

Python tuples are identified by "()" and internal elements are separated by ":".

A. True

B. False

ANSWER: B

Explanation:

False is correct because Python tuple elements are separated by commas, not colons. Parentheses are commonly used to make a tuple's grouping explicit, for example `(1, 2, 3)`, but the comma is the syntax that actually defines a tuple. This is especially clear for a one-element tuple: it must be written as `(1,)`; writing `(1)` is simply an integer in parentheses. Parentheses can also be omitted in many contexts, such as `x = 1, 2`, which assigns the tuple `(1, 2)` to `x`. Colons have other Python roles, including separating dictionary keys from values and defining slices, but they do not delimit tuple elements. Therefore, the statement is false because it incorrectly states that tuple members are separated with colons. The Python language reference describes tuple displays as comma-separated expressions and notes that the comma, rather than the parentheses, is what forms a tuple. See the Python language reference on displays and the Python standard types documentation for tuples.

QUESTION NO: 30

The following about the standard RNN Model, the correct statement is?

A. There is no one-to-one model structure

B. Do not consider the time direction when backpropagating

C. There is no many-to-many model structure

D. There will be a problem of attenuation of long-term transmission and memory information

ANSWER: D

Explanation:

There will be a problem of attenuation of long-term transmission and memory information is correct. A standard recurrent neural network propagates information through a hidden state over successive time steps. During training, backpropagation through time repeatedly applies derivatives across that temporal chain. When these derivatives are commonly smaller than one, their product can shrink exponentially as the distance between the relevant earlier input and the current loss increases. This is the vanishing-gradient problem: learning signals for distant time steps become too weak for the network to reliably retain or learn long-range dependencies. In practical sequence tasks, this can make a conventional RNN favor recent context and lose information that must persist over many steps. The issue is a foundational limitation of ordinary RNNs and was a key motivation for gated recurrent architectures, such as long short-term memory networks, whose gated memory path helps preserve gradients and information over longer intervals. The original LSTM work explicitly addresses long time lags in recurrent learning; see Long Short-Term Memory. For a clear technical discussion of recurrent neural networks, backpropagation through time, and vanishing gradients, see Deep Learning, Chapter 10: Sequence Modeling.

QUESTION NO: 31

In random forest, what strategy is used to determine the outcome of the final ensemble model?

A. Cumulative system

B. Find the average

C. Voting system

D. Cumulative system

ANSWER: B C

Explanation:

Random forest combines the predictions produced by many independently built decision trees to obtain a final ensemble result. The applicable aggregation strategy depends on the learning task. **Voting system** is used for classification: each tree predicts a class, and the forest selects the class receiving the most votes. This majority-voting approach reduces the influence of errors made by individual trees and generally produces a more stable classifier. **Find the average** is used for regression: each tree outputs a numeric prediction, and the forest computes the mean of those outputs as its final prediction. Averaging reduces variance and makes the regression estimate less sensitive to any one tree or to variation introduced

during training. Therefore, both strategies describe how a random forest determines its final output, with voting corresponding to classification and averaging corresponding to regression. This distinction is reflected in standard random-forest implementations: scikit-learn documents that its classifier predicts using the mean predicted class probabilities across trees, while its regressor predicts by averaging tree predictions. See the scikit-learn forest ensemble documentation and the RandomForestRegressor API reference.

QUESTION NO: 32

According to the development process of the robot, it is usually divided into three generations, respectively are: (Multiple Choice)

A. Teaching Reproduction Robot

B. Robot with sensation

C. Robots that will think

D. Intelligent robot

ANSWER: A B D

Explanation:

The conventional three-generation classification of robots is based on the progressive growth of control capability, environmental perception, and autonomous decision-making. **Teaching Reproduction Robot** represents the first generation: an operator teaches positions and motion sequences, which the robot subsequently stores and repeats. This approach remains a foundational concept in industrial robot programming and repeatable manufacturing tasks. **Robot with sensation** represents the second generation, in which sensors provide feedback about conditions such as position, force, vision, or proximity. Sensor input allows the robot to adapt its actions to changes in its working environment rather than merely replaying a fixed trajectory. **Intelligent robot** represents the third generation, combining sensing with higher-level information processing, planning, learning, and decision-making so that the system can perform more autonomously in complex or less structured environments. This progression—from programmed repetition, through feedback-based perception, to intelligent autonomy—is a commonly used educational model for describing robot evolution. General robotics references describe robots as systems combining sensing, control, and action, while industrial robotics resources document the importance of programmable and sensor-enabled automation. See Encyclopaedia Britannica: Robot technology and the International Federation of Robotics: Industrial Robots.

QUESTION NO: 33

Which of the following is not MindSpore common Operation?

A. signal

B. math

C. nn

D. array

ANSWER: A

Explanation:

signal is the correct choice because it is not one of the standard “common operation” categories used in MindSpore’s operation taxonomy. MindSpore organizes its built-in operators into functional groups that include array-oriented operations, mathematical operations, and neural-network operations. These groupings help developers locate operators according to the type of tensor processing or model computation they perform. For example, array operations cover tensor shape, indexing, concatenation, and similar structural processing; mathematical operations provide numerical and algebraic computation; and neural-network operations provide primitives used to construct and execute deep-learning layers and related model functions. The MindSpore API documentation presents operators under these established operation groupings rather than

defining a general common-operations category named `signal`. Although signal-processing functionality can be implemented with available tensor and mathematical primitives, “signal” is not a common-operation classification in the terminology assessed by this question. See the MindSpore ops API reference and the MindSpore operations reference for the documented operator organization.

QUESTION NO: 34

HUAWEI HiAI Engine Can easily combine multiple AI Ability and App integrated.

A. TRUE

B. FALSE

ANSWER: A

Explanation:

TRUE is correct. Huawei HiAI is an AI computing platform designed to make AI capabilities available to applications through an integrated framework. Its purpose is not limited to running one isolated model; it enables developers to access and combine available AI services and device-side AI capabilities within an app. Typical capabilities made available through the HiAI ecosystem include computer vision, speech, natural-language processing, and other intelligent processing functions. By exposing these capabilities through development tools and APIs, the platform reduces the work required for an application to incorporate more than one AI function and supports integrated intelligent experiences. This aligns with Huawei’s description of HiAI as a mobile AI computing platform that provides open AI capabilities and supports application development across hardware and software resources. The wording “combine multiple AI Ability and App integrated” is grammatically imperfect, but it conveys the intended platform feature: multiple AI abilities can be integrated into an application to deliver richer functions. Huawei’s HiAI overview is available at Huawei Developer: HiAI, and Huawei’s developer portal provides the broader AI development ecosystem and documentation at Huawei Developer.

QUESTION NO: 35

Which of the following is not an artificial intelligence school?

A. Symbolism

B. Statisticalism

C. Behaviorism

D. Connectionism

ANSWER: B

Explanation:

Statisticalism is the correct selection because the standard introductory classification used in Huawei AI learning materials identifies symbolism, connectionism, and behaviorism as the principal schools of artificial intelligence. Symbolism approaches intelligence through explicit knowledge representation, symbols, and logical reasoning. Connectionism models intelligent behavior through interconnected processing units, forming the conceptual basis for neural-network approaches. Behaviorism emphasizes an agent’s interaction with its environment and learning from feedback, which is closely associated with reinforcement-learning concepts. These three labels are therefore recognized as established AI schools in this classification. “Statisticalism” is not included as one of those named schools. Although modern AI frequently uses statistical methods, probability, and data-driven learning, that does not make “Statisticalism” a school in this particular canonical three-school taxonomy. Huawei describes AI as encompassing technologies such as machine learning and neural networks in its AI overview: Huawei: Artificial Intelligence. For broader background on the historical and conceptual foundations of AI, see the Stanford Encyclopedia of Philosophy’s overview: Artificial Intelligence.

QUESTION NO: 36

Which of the following statements about Python are correct? (Multiple choice)

A. Invented m 1989 by the Dutch Guido van Rossum, the first public release was issued in 1991.

B. Python 1s purely free software and the source code follows the GPL (GNU General Public License) protocol

C. Python syntax is simple and clean One of the features is to force blank characters to be used as statement 1ndentat1on [Right Answers)

D. Python 1s often nicknamed glue language which can easily connect various modules made 1n other languages

ANSWER: A C D

Explanation:

Python was created by Guido van Rossum, with implementation work beginning in 1989 and the first public release appearing in 1991. This origin is recorded in the Python documentation's historical material and remains a central fact about the language's development. Python also deliberately uses indentation to define the block structure of compound statements such as functions, classes, loops, and conditional statements. The language reference specifies that leading whitespace determines indentation levels, making visual structure part of Python syntax rather than merely a formatting convention.

Python is also widely characterized as a "glue language." Its straightforward extension and embedding capabilities, together with interfaces for calling native libraries and interoperating with code written in languages such as C and C++, make it practical for connecting components in larger systems. This role is particularly valuable in AI and data-processing workflows, where Python often orchestrates high-performance libraries and services implemented elsewhere. See the Python language reference on indentation and the Python documentation on extending and embedding.

QUESTION NO: 37

Huawei AI The full scenarios include public cloud, private cloud, various edge computing, IoT industry terminals, and consumer terminals and other end, edge, and cloud deployment environments.

A. TRUE

B. FALSE

ANSWER: A

Explanation:

TRUE is correct. Huawei describes its AI strategy as providing full-stack, all-scenario AI capabilities, with deployment spanning cloud, edge, and devices. This approach is intended to let organizations use AI where data is generated and where inference or training is best suited: centralized cloud environments can provide large-scale computing and model development resources; edge environments can support lower-latency processing nearer to data sources; and device or terminal environments can enable intelligence directly on consumer devices and industry IoT endpoints. The statement's reference to public cloud, private cloud, edge computing, industry terminals, and consumer terminals is therefore consistent with Huawei's end-edge-cloud AI deployment model. Huawei's AI materials describe this all-scenario vision as covering devices, edge infrastructure, and cloud platforms rather than limiting AI to a single centralized environment. This broad deployment coverage supports different operational, latency, connectivity, and data-governance requirements across industries. See Huawei's overview of its AI strategy and capabilities and its enterprise AI solutions.

QUESTION NO: 38

In the face search service, if we want to delete a certain face set, we can use this code:

firs_cliet.get_v2().get_face_set_service().delete_face_set("**"), among them "**" is to fill in the actual face set name.

A. TRUE

B. FALSE

ANSWER: A

Explanation:

TRUE is correct. The Face Recognition Service SDK exposes face-set operations through the versioned face-set service obtained from the initialized client. Its `delete_face_set` operation accepts the target face-set name, so supplying the actual name in place of `"**"` identifies the face set to be removed. In practical code, the client variable must already have been created and authenticated with credentials that permit face-set deletion. The operation is directed at the named face set rather than at an individual face record; deleting a face set removes the collection itself and therefore should be used only after confirming that this is the intended resource. This SDK call follows the normal Huawei Cloud SDK pattern of obtaining a service interface from a versioned client and invoking the resource-management method with the required identifier. Huawei's Python SDK repository provides the SDK implementation and service modules at HuaweiCloud SDK for Python. The corresponding REST operation and its required face-set identifier are documented in the Face Recognition Service API Reference.