

ArcGIS Desktop Associate

Esri EADA10

Version Demo

Total Demo Questions: 8

Total Premium Questions: 87

Buy Premium PDF

<https://passqueen.com>

support@passqueen.com

Topic Break Down

Topic	No. of Questions
Topic 1, Data Management	46
Topic 2, Visualization	16
Topic 3, Editing	7
Topic 4, Analysis	12
Topic 5, Sharing	6
Total	87

QUESTION NO: 1

Which ArcGIS Desktop extension can be used to create a map with advanced label placement options?

- A. Network Analyst
- B. Schematics
- C. 3D Analyst
- D. Publisher
- E. Maplex

ANSWER: E

Explanation:

Maplex is the ArcGIS Desktop labeling extension designed for advanced label placement and cartographic-quality output. When the Maplex Label Engine is enabled, it provides more sophisticated rules for placing labels than the standard labeling engine, including improved conflict detection, label stacking, abbreviation strategies, placement around features, and controls for resolving label overlaps. These capabilities are especially useful for dense or complex maps where labels need to remain readable while clearly associated with their features. In ArcMap, the extension is enabled through the Extensions dialog, and the Maplex Label Engine can then be selected from the Labeling toolbar or labeling settings. The engine supports specialized placement properties for point, line, and polygon features, allowing a map author to refine the layout without manually converting every label to annotation. Esri describes Maplex as an extension that provides enhanced label-placement capabilities for producing high-quality cartographic maps. See Esri's [Maplex for ArcGIS overview](#) and [Maplex Label Engine documentation](#).

QUESTION NO: 2

Which element is required for creating Data Driven Pages?

- A. page number
- B. sort field
- C. spatial reference
- D. rotation
- E. index layer

ANSWER: E

Explanation:

An index layer is required for creating Data Driven Pages. In ArcMap, Data Driven Pages use the features in an index layer to define the geographic extent of each map page. Each index feature represents one page, and ArcMap iterates through those features to generate the map series. When enabling Data Driven Pages from the Data Driven Pages Setup dialog box, the user must select both the data frame that will be driven and an index layer from that data frame. The index layer can be a polygon layer for areas such as counties or map-grid cells, but it may also use points or lines depending on the desired map-series behavior. Attributes in the index layer can optionally supply page names, page numbering, sorting, and rotation values, but those are not fundamental prerequisites. The essential concept is that the index layer supplies the individual features and extents that make the pages data driven. Esri's ArcMap documentation describes the index layer as the layer used to define the geographic extent for each page in the series. See [What are Data Driven Pages?](#) and [Setting up Data Driven Pages](#).

QUESTION NO: 3

Which two actions taken by an ArcGIS user result in conflicts being detected during a reconcile operation? (Choose two)

- A. Changing the same feature in two separate versions of the geodatabase
- B. Changing the same feature in the same version of the geodatabase in different edit sessions
- C. deleting a feature in a child version of the geodatabase, but not in the parent version
- D. deleting a feature in one edit session and altering it in another inside the same version of the geodatabase
- E. changing a feature in a child version of the geodatabase and deleting it in the parent version

ANSWER: A E

Explanation:

Conflicts can be detected when edits in the version being reconciled overlap with edits that were made in its target version after the child version was created or last reconciled. “Changing the same feature in two separate versions of the geodatabase” produces a conflict when those versions participate in the reconcile relationship, because both versions have modified the same row independently. With the default row-level conflict definition, ArcGIS identifies that row as having competing edits and presents it for conflict review during reconciliation.

“Changing a feature in a child version of the geodatabase and deleting it in the parent version” is also a reconcile conflict. The child contains an update to the row while the target version contains a deletion of that same row. ArcGIS recognizes this update-versus-delete condition as conflicting concurrent edits. The conflict-detection process compares edits in the child version with edits posted to the target version since the versions shared a common state; it is not simply a check for whether a feature was edited at some point. For details on the conflict types and detection behavior in traditional versioning, see [ArcGIS Pro: Conflict detection in versioning](#) and [ArcGIS Pro: Reconcile and post edits](#).

QUESTION NO: 4

A coded value domain provides which two benefits for attribute editing? (Choose two)

- A. It specifies valid values for an attribute
- B. It stores rules for attribute display
- C. It displays a user-friendly description
- D. It validates all attributes in a given table
- E. It specifies valid values for an attribute and displays a user-friendly description

ANSWER: E

Explanation:

It specifies valid values for an attribute and displays a user-friendly description is correct because a coded value domain defines a fixed, allowable list of codes for a field. During editing, this limits entered values to the values approved for that attribute, promoting consistent data entry and protecting attribute integrity. Each stored code can also have a descriptive name. Editors can select and view the meaningful description in the editing interface, while the underlying attribute value remains the defined code. For example, a field may store short values such as *R*, *C*, and *I*, while presenting descriptions such as Residential, Commercial, and Industrial to the editor. This pairing makes data entry both controlled and easier to understand, especially when stored codes are abbreviated or numeric. Coded value domains are created at the geodatabase level and can be assigned to compatible fields, providing the same valid-value list and descriptions wherever the domain is used. Esri documents that coded value domains enforce a set of valid values and that each code has a corresponding description. See [Esri's overview of attribute domains](#) and [Esri's documentation on creating and managing domains](#).

QUESTION NO: 5

An ArcGIS user is preparing a street layer for a small-scale regional map. Only streets with a CLASS value of LOCAL should be displayed, and those streets should not draw when the map is zoomed out beyond 1:24,000.

Which two layer settings should the user configure? (Choose two)

- A. Set a definition query for CLASS = 'LOCAL'.
- B. Set a scale range that hides the layer when zoomed out beyond 1:24,000.
- C. Select all records with CLASS = 'LOCAL'.
- D. Set layer transparency to 100 percent beyond 1:24,000.
- E. Create a relate between the street layer and the map document.

ANSWER: A B

Explanation:

A **definition query** should be used to restrict the layer to records where the CLASS field has the value LOCAL. Features that do not satisfy the query are not displayed or made available through the layer.

A **scale range** should be used to prevent the layer from drawing beyond the specified scale. This supports a clearer regional map by avoiding dense local-street detail at small scales. Selecting features or applying transparency does not filter the layer by attribute value and scale in the required way.

QUESTION NO: 6

An ArcGIS user would like to share an .mxd file with another co-worker without impacting the symbology of the layer.

Which two files can be sent to ensure the symbology is maintained? (Choose two)

- A. layer package file
- B. representation file
- C. .lyr file
- D. style file
- E. symbology file

ANSWER: A C

Explanation:

layer package file and **.lyr file** preserve the layer's symbology and are appropriate formats for sharing configured ArcMap layers. A layer file stores a layer's properties, including its symbology, labeling, definition query, joins, and other display settings. When the recipient adds the layer file in ArcMap, the layer can retain the cartographic appearance configured by the sender, provided that the referenced data remains accessible. A layer package is a portable package that includes the layer file plus the data used by the layer, making it especially useful when the coworker may not have access to the sender's original data paths. Because the package carries both the data and the layer's saved properties, it helps reproduce the intended display and symbology on another machine. Esri documents that layer files save layer properties and that layer packages package layers with their data for sharing. See [Esri: About layer files](#) and [Esri: About layer packages](#).

QUESTION NO: 7

A map document references a local data source.

What must an ArcGIS user do before sending the map document to another workstation? (Choose two)

- A. set the .mxd file to store full paths
- B. provide the associated data referenced by the map document

- C. update metadata for the map document
- D. set the .mxd file to store relative paths

ANSWER: B D

Explanation:

The correct preparation is to provide the associated data referenced by the map document and to set the .mxd file to store relative paths. An ArcMap map document stores references to datasets, such as feature classes, shapefiles, rasters, and tables; it does not embed those local source datasets in the .mxd. Therefore, the recipient needs access to copies of the referenced data, ideally organized in a folder structure that accompanies the map document. Relative paths record each data location in relation to the folder containing the map document rather than using a workstation-specific drive letter and absolute directory path. When the map document and its data are copied together while preserving that relative structure, ArcMap can resolve the data references on the receiving workstation without requiring the same user name, drive letter, or root folder location. Esri documents that map documents can store relative path names and that this is useful when moving a map and its data to another location. For a more managed transfer workflow, a map package can also consolidate the map and its referenced data. See [Esri: About map document paths](#) and [Esri: Creating map packages](#).

QUESTION NO: 8

What is the result of applying a relate in ArcMap to a stand-alone table and a layer attribute table?

- A. It will define the relationship between them
- B. It will append the associated data to the layer's attribute table
- C. It will append the layer's attribute table to the associated data
- D. It will create a new feature class containing the layer's attributes and associated data

ANSWER: A

Explanation:

It will define the relationship between them. In ArcMap, a relate establishes a temporary association between records in a layer's attribute table and records in a stand-alone table using matching key-field values. Unlike a join, a relate does not merge, append, or physically copy fields from one table into the other. Each table remains separate, which is particularly useful when one feature may be associated with multiple records in the related stand-alone table, such as a parcel linked to several inspection or ownership-history records. After the relate is created, users can select a record and use the related-data commands to view the matching records in the other table. The relate is stored with the map document rather than changing the source datasets themselves, so it provides a way to navigate and query associated records while preserving the original table structures. Esri distinguishes relates from joins specifically because a relate defines this association without adding fields to the layer's attribute table. See Esri's [documentation on relates](#) and its overview of [joining and relating tables](#).