**Course Name: .Net 6 Core Entity Framework: How-To Guide For Professionals**

*NOTE: Mark the correct answers with Yellow highlight*

**Chapter Number 3 – Fluent API**

1. The Fluent API is used for…

a. Providing in-line documentation and help text.

Reason – incorrect, they are not used for either of those purposes.

b. describing the data types used to map to class properties.

Reason – Correct, they are used to refine the physical storage for class properties; this includes the column name, data storage structure and database-specific characteristics (depending on the type of database.

c. establishing relationships between classes.

Reason – Incorrect, relationships are just one aspect of class-models that can be described using data annotations.

d. Pulling database schema definition into class models.

Reason – Incorrect, Migrations are used to move models into the database; another technique is used to move database schema elements into models (re: Scaffolding).

2. What types of relationship can be described with Data Annotations?

a. Parent-Child and Child-Sibling.

Reason – Incorrect, these are not relationship types.

b. Many-To-One and One-To-Many.

Reason – Incorrect, Many-to-One and One-to-Many are two ways to describe only one type of relationship supported.

c. One-to-One, One-to-Many, and Many-to-Many.

Reason – Correct, Data Annotations can be used to model all types of class relationships.

d. Design-Time and Run-Time.

Reason – Incorrect, Design-Time and Run-Time are types of ***DbContext*** instances used by the .net entity framework (the former is used by migrations to define the changes needed to the database schema, the latter by the application to provide instance data to the end-user or a process).

3. How do you specify the primary key property?

a. add [PrimaryKey] annotation to property definition.

Reason – Incorrect, there is no data annotation defined as “PrimaryKey”.

b. add \_PK to the name of the property.

Reason – Incorrect, this will do nothing but change the name of the property.

c. by convention, using a column name “Id” or “ClassNameID”, or using Data Annotation “Key”.

Reason – Correct, you can rely on the default behavior (convention) or use the “Key” Data Annotation on the class property definition.

d. Add [KEYNAME=’property’] as a programming directive to the class definition.

Reason – Incorrect, there is no such programming directive.

4. What does Data Attribute “Not Mapped” mean?

a. Create the property in the under databased schema, but do not map it.

Reason Incorrect – you cannot use data annotations to achieve this.

b. The underlaying database table has no primary key.

Reason – Incorrect, totally made up answer.

c. The property can be assigned a value and returned, but it will not be persisted in the database and has no related storage space

Reason – Correct, the class property value will only exist for the lifetime of the class as it is not saved to the database (has no physical storage).

d. The database will create a custom query plan for every data access request by the class.

Reason – Incorrect; Data Annotation have no impact on the query plans created by a database.

5. What are the possible Primary Key value definitions?

a. Default, Sequential, Calculated, and None.

Reason – Incorrect; Identity is the default (not “Default”), Sequential does not exist (identity is sequential, but can have gaps where rows have been deleted).

b. Identity, GUID, Computed, and None.

Reason – Correct; these are the 4 possible Primary Key values, where Identity is the default.

c. Identity, Sequential, None.

Reason – Incorrect; Sequential does not exist, missing GUID and Computed.

d. Identity, GUID, Calculated, and None.

Reason – Incorrect; Calculated is not known type (Computed is the correct term).