Ankita

INTERVIEW QUESTIONS ON APEX CLASS SALESFORCE

1. What is Apex?

Answer:
Apex is a strongly typed, object-oriented programming language that is used to execute flow and transaction control statements on the Salesforce platform's server. It is primarily used to write custom logic such as triggers, classes, and asynchronous operations.

2. What is the difference between a Trigger and a Class in Apex?

Answer:

- **Trigger**: A trigger is used to automatically perform actions when data is inserted, updated, deleted, or undeleted in Salesforce objects.
- **Class**: A class is a container for methods, variables, and logic that can be called from triggers, Visualforce pages, or other classes.

3. What is the syntax for defining a class in Apex?

Answer:

```
apex
public class MyClass {
    // Class members
}
```

4. What is the purpose of the `with sharing` and `without sharing` keywords in Apex?

Answer:

- **with sharing**: Enforces the sharing rules of the current user's profile. The class will respect the visibility of records based on sharing settings.
- **without sharing**: Ignores the sharing rules and allows the class to operate without regard to record visibility.

5. What are Apex Governor Limits?

Answer:
Governor limits are limits enforced by the Salesforce platform to ensure fair use of shared resources. These limits include limits on the number of queries, records processed, and the total heap size. For example, a trigger can run a maximum of 100 SOQL queries.

6. What is a Static Variable in Apex?

Answer:
A static variable in Apex is shared across all instances of a class. It is initialized only once and retains its value across method calls. It is often used to maintain state across different transactions in a class.

7. What is the difference between `public`, `private`, and `protected` access modifiers in Apex?

Answer:

- **public**: The class or method can be accessed from any class.
- **private**: The class or method can only be accessed within the same class.
- **protected**: The class or method can be accessed within the same class and by subclasses.

8. What is the `@future` annotation in Apex?

Answer:
The `@future` annotation is used to execute methods asynchronously, outside the scope of the current transaction. It is typically used for operations that are time-consuming, like sending email notifications or making HTTP requests.

Example:

```
apex
@future
public static void sendEmail(String emailId) {
    // Logic to send an email
}
```

9. What are Apex Collections?

Answer:
Apex collections are used to store multiple values. The main types of collections in Apex are:

- **List**: An ordered collection of elements, can contain duplicates.
- **Set**: An unordered collection of unique elements.
- **Map**: A collection of key-value pairs.

10. What is a Constructor in Apex?

Answer:
A constructor is a special method in a class used to initialize objects. It is called when an object of the class is instantiated. Apex provides a default constructor and allows the creation of custom constructors with parameters.

Example:

```
apex
public class MyClass {
    public String name;

    // Custom constructor
    public MyClass(String name) {
        this.name = name;
    }
}
```

11. What is the `transient` keyword in Apex?

Answer:
The `transient` keyword is used to mark instance variables in a class that should not be serialized. This is useful when we do not want certain data to be persisted or saved when the object is passed through the Visualforce page, especially when dealing with large data objects.

12. What is the purpose of the `try-catch` block in Apex?

Answer:
The `try-catch` block is used for exception handling in Apex. The `try` block contains code that may throw exceptions, and the `catch` block contains code to handle those exceptions. You can also use the `finally` block to run cleanup code regardless of whether an exception occurs.

Example:

```
apex
try {
    // code that may throw exception
} catch (Exception e) {
    // handle exception
} finally {
    // cleanup code
}
```


13. What is the difference between `SOQL` and `SOSL` in Apex?

Answer:

- **SOQL (Salesforce Object Query Language)**: Used to query a single object or related objects. It is used to retrieve records based on certain criteria.
- **SOSL (Salesforce Object Search Language)**: Used to search across multiple objects for a given search term.

Example:

- **SOQL**: `SELECT Name FROM Account WHERE Industry = 'Technology'`
- **SOSL**: `FIND 'Acme' IN ALL FIELDS RETURNING Account(Name), Contact(Name)`

14. How can you avoid hitting governor limits in Apex?

Answer:
To avoid hitting governor limits, you can:

- Use batch processing for large data operations.
- Minimize the number of queries by using relationship queries and efficient filters.
- Use `Collections` like Lists, Sets, and Maps to minimize the number of DML statements and queries.
- Ensure that triggers are bulkified (handle multiple records at once).
- Implement proper error handling and logging.

15. What is Batch Apex in Salesforce?

Answer:
Batch Apex is used to process large amounts of data asynchronously. It divides the data into smaller chunks, processes each chunk separately, and then completes the entire job. Batch Apex is typically used when the amount of data exceeds the governor limits.

Example:

```
apex
global class MyBatchClass implements Database.Batchable<SObject> {
    global Database.QueryLocator start(Database.BatchableContext BC) {
        return Database.getQueryLocator([SELECT Id FROM Account]);
    }

    global void execute(Database.BatchableContext BC, List<Account> scope)
{
        // Processing logic
    }

    global void finish(Database.BatchableContext BC) {
        // Finalize logic
    }
}
```

16. What are the different types of Apex Tests?

Answer:
Apex provides three types of test classes:

- **Unit Tests**: These test individual units of code for correctness.
- **Integration Tests**: These test interactions between different components.
- **End-to-End Tests**: These test the system as a whole, ensuring all components work together.

Apex tests are crucial for ensuring code quality and that Apex code behaves as expected in different scenarios.