12. Creating an Application in Salesforce.com Using Apex Programming Language

Perform Subtraction of Two Numbers

- Step 1: Log in to Salesforce

Open https://login.salesforce.com

Enter your username and password

Click Log In

- Step 2: Open Developer Console

Click the gear icon (⚙️) at the top-right corner

Select Developer Console

- Step 3: Create a New Apex Class

In Developer Console: File > New > Apex Class

Name the class: SubtractionApp

Paste the following code:

```
public class SubtractionApp {

    // Method to subtract two numbers
    public static Integer subtract(Integer a, Integer b) {
        return a - b;
    }
```

}

Click OK to save

- Step 4: Run Code Using Execute Anonymous Window

In Developer Console: Debug > Open Execute Anonymous Window

Paste this test code:

```
Integer result = SubtractionApp.subtract(15, 7);
System.debug('Result: ' + result);
```

Check “Open Log” (optional)

Click Execute

- Step 5: View the Output

Go to the Logs tab in Developer Console

Open the latest log

Press Ctrl + F and search: Result:

You should see:

```
USER_DEBUG [2]|DEBUG|Result: 8
```

Theory

🔹 Apex Class:

```
public class SubtractionApp {

    // Method to subtract two numbers
    public static Integer subtract(Integer a, Integer b) {
        return a - b;
    }
}
```

🔹 Execute Anonymous Code (to run and test the class):

```
Integer result = SubtractionApp.subtract(15, 7);
System.debug('Result: ' + result);
```

📘 Detailed Line-by-Line Explanation

Line	Explanation
public class SubtractionApp	Declares a public class that contains the logic for subtraction. Classes organize code logically in Apex.

public static Integer subtract(Integer a, Integer b)

Defines a static method that takes two integers, subtracts them, and returns the result. Static methods can be called without instantiating the class.

return a - b;

Subtracts the second number from the first and returns the result.

System.debug('Result: ' + result);

Outputs the result to the Debug Logs for testing and verification.

💡 Theory for Viva – Apex, Salesforce, and Cloud

🔹 1. What is Salesforce?

Answer:Salesforce is a cloud-based CRM (Customer Relationship Management) platform that helps businesses manage customer data, automate processes, and develop custom applications using low-code and code-based tools like Apex.

🔹 2. What is Apex in Salesforce?

Answer:Apex is a strongly-typed, object-oriented programming language for Salesforce, similar to Java. It is used to write business logic, triggers, web services, scheduled jobs, and custom controllers.

🔹 3. Where is Apex code executed?

Answer:Apex code runs on Salesforce's multi-tenant cloud platform, specifically the Lightning Platform (formerly Force.com), in a hosted environment.

🔹 4. How do you run Apex code?

Answer:Apex code can be executed in several ways:

Execute Anonymous Window in Developer Console for testing.

Triggers for automating data operations.

Classes and Methods for reusable logic.

Visualforce or Lightning Components for UI-based applications.

- 5. What is the Developer Console in Salesforce?

Answer:The Developer Console is a browser-based IDE in Salesforce for:

Writing and executing Apex code.

Debugging with logs.

Running SOQL queries.

Managing and running tests.

- 6. What are Static Methods in Apex?

Answer:Static methods:

Belong to the class, not an instance of the class.

Can be called directly using ClassName.methodName().

Are used for utility functions, like the subtraction method in this example.

- 7. What is System.debug()?

Answer:System.debug() is used to:

Print output to the Debug Logs during development.

Assist in troubleshooting and testing.

Display information only to developers, not end users.

🔹 8. What are some benefits of using Apex?

Tight integration with Salesforce data (objects, fields).

Built-in security and governor limits.

Compatibility with Lightning and Visualforce.

Enables automation of complex business logic.

🔹 9. What are Governor Limits in Apex?

Answer:Governor Limits are Salesforce's restrictions to ensure fair resource usage in its multi-tenant environment, such as:

CPU Time: Max 10,000 ms.

SOQL Queries: Max 100 per transaction.

DML Statements: Max 150 per transaction.

Heap Size: Max 6 MB.These limits prevent any single process from monopolizing server resources.

🔹 10. What is the Force.com Platform (Lightning Platform)?

Answer:The Lightning Platform is Salesforce's PaaS (Platform-as-a-Service) layer, enabling:

Custom application development.

Business logic creation using Apex.

Custom UI development with Visualforce or Lightning Components.

📌 Summary

Component	Description
Apex	Programming language for Salesforce platform.
Static Class Method	Subtracts two numbers and returns the result.
System.debug()	Outputs results to Debug Logs for testing.
Developer Console	Tool for writing, running, and testing Apex code.
Salesforce Platform	Cloud-based CRM and app development platform.

🧭 Salesforce and Apex Concepts

🧠 1. What is Salesforce.com?

Salesforce.com is a cloud-based CRM platform offering:

SaaS: Pre-built applications like Sales Cloud and Service Cloud.

PaaS: Development tools via the Lightning Platform for custom apps.

🧠 2. What is Apex Programming?

Apex is:

A server-side, Java-like programming language for Salesforce.

Designed for interacting with Salesforce data.

Used for custom logic, triggers, web services, scheduled tasks, and batch processing.

Compiled and executed entirely in the Salesforce cloud.

🧰 3. Where is Apex Used in Salesforce?

Component

Purpose

Triggers

Automate data operations (insert, update, delete).

Classes & Methods

Reusable logic (e.g., SubtractionApp).

Web Services

Expose logic to external systems.

Scheduled Jobs

Run tasks at specific times.

Batch Apex

Process large datasets.

Visualforce/Lightning

Backend logic for user interfaces.

🔗 4. How Does the Code Fit in Salesforce Architecture?

The SubtractionApp Apex class is:

Compiled and stored on Salesforce servers.

Executable via:

Developer Console’s Execute Anonymous Window.

UI components (Visualforce or Lightning).

Automation tools (Flow, Process Builder, or REST API).

🛠️ 5. What Is the Developer Console?

The Developer Console is an in-browser IDE for:

Writing and running Apex code.

Viewing Debug Logs.

Running SOQL queries.

Analyzing performance and running tests.

🔁 6. How Does Code Execute? (Lifecycle)

Write the Apex class (SubtractionApp).

Compile and save it to Salesforce’s cloud.

Run the code via the Execute Anonymous Window.

Use System.debug() to output results to the Debug Log.

View the results in the Logs tab.

📊 7. What is the Role of Static Methods in This Code?

Static methods:

Allow calling the subtract method without instantiating SubtractionApp.

Are memory-efficient for utility tasks like subtraction.

Simplify code reuse.

🔐 8. What are Governor Limits, and Why Do They Matter?

Governor Limits ensure fair resource usage in Salesforce's multi-tenant environment. Examples include:

DML Statements: Max 150 per transaction.

SOQL Queries: Max 100 per transaction.

Heap Size: Max 6 MB.

CPU Time: Max 10,000 ms.This simple subtraction code stays well within these limits.

⚙️ 9. Advantages of Using Apex

Deep integration with Salesforce data.

Built-in security (e.g., user and object-level permissions).

Native cloud execution with auto-scaling.

Compatibility with Salesforce features like Flows and Process Builder.

💡 10. How Can This Code Be Extended for Real-World Apps?

The SubtractionApp class can be extended to:

Accept user input via Visualforce or Lightning forms.

Store results in a custom Salesforce object.

Run in a trigger to calculate field differences.

Be exposed via a REST API for external use.

Integrate with Flows for automated field calculations.

🧪 Sample Viva Questions with Smart Answers

Question

Suggested Answer

What is Apex?

Apex is Salesforce's object-oriented programming language for custom logic, triggers, and automation.

Where can we run Apex code?

In triggers, classes, Developer Console, Visualforce, Lightning Components, or via APIs.

What is the platform on which Apex runs?

The Lightning Platform (formerly Force.com), Salesforce's PaaS for custom app development.

Why use static methods here?

Static methods simplify calling utility functions like subtraction without creating objects.

How does this code execute in Salesforce?

It's compiled, stored in the Salesforce cloud, and executed on-demand via the Developer Console or other mechanisms.