

Development Scenario 2: Event Management System

Day 1: Introduction and Setup

Task 1: Set up the Kotlin development environment and write a simple Kotlin script to validate the setup.

Step 1: Install Java Development Kit (JDK)

1. Download JDK.
2. Set up JAVA_HOME environment variable.
3. Update PATH environment variable.
4. Verify Installation

Step 2: Install Kotlin

1. Download Kotlin Compiler (kotlinc).
2. Extract Kotlin Compiler.
3. Set up PATH environment variable for Kotlin.

Step 3: Write and Run a Simple Kotlin Script

1. Create a Kotlin script:
 - Create a sample program with extension '.kt',v file name "first.kt".
 - Program : fun main(){
 Println("Hello Marvel")
 }
2. Save Script: Select the location where the program want to be saved.
3. Compile And run the Kotlin Script:
 - Open Command Prompt and navigate to the directory where 'first.kt' is saved.
 - Compile the Kotlin script using kotlinc first.kt -include-runtime -d Hello.jar.
 - Run the compiled Kotlin program using java -jar first.jar.
4. Output: Hello Marvel

Task 2: Experiment with Kotlin's string templates to create dynamic welcome messages.

File name : Message.kt

Program:

```
import java.util.Scanner;

fun main() {

    val scanner = Scanner(System.in)

    print("Enter your first name: ")

    val firstName = scanner.nextLine()

    print("Enter your last name: ")

    val lastName = scanner.nextLine()
}
```

```
        val message = "Welcome, $firstName $lastName."
        println(message)
    }
```

Compile and Run: Save program with Message.kt.

Output: Welcome Vin Kora

Task 3: Define data types to represent event details such as name, date, and attendee count.

File Name: EventDetails.kt

Program:

```
class EventDetails(
    val eventName: String,
    val eventDate: String,
    val attendeeCount: Int
)

fun main() {
    val event = EventDetails("Birthday Party", "18-07-2024", 20)
    println("Event Name: ${event.eventName}")
    println("Event Date: ${event.eventDate}")
    println("Attendee Count: ${event.attendeeCount}")
}
```

Output: Event Name: Birthday Party

Event Date: 18-07-2024

Attendee Count: 20

Task 4: Implement a basic user input flow to create new events using if and when statements.

File Name: Event.kt

Program: data class Event{

```
    val eventName: String,
    val eventDate: String,
    val attendeeCount: Int
}
```

```

//Main Program
import java.util.Scanner

fun main() {
    val scanner = Scanner(System.`in`)
    println("Welcome to the Event Manager!")
    var events = mutableListOf<Event>()
    var running = true
    while (running)
    {
        println("\nSelect an action:")
        println("1. Create a new event")
        println("2. View all events")
        println("3. Exit")
        print("Enter your choice: ")
        val choice = scanner.nextInt()
        when (choice) {
            1 -> {
                println("\nEnter event details:")
                print("Event Name: ")
                scanner.nextLine()
                val eventName = scanner.nextLine()
                print("Event : ")
                val eventDate = scanner.nextLine()
                print("Attendee Count: ")
                val attendeeCount = scanner.nextInt()
                val newEvent = Event(eventName, eventDate, attendeeCount)
                events.add(newEvent) println("Event created successfully!")
            }
            2 -> {
                if (events.isEmpty())
                {

```

```

        println("\nNo events created yet.")
    }
    else {
        println("\nAll Events:")
        events.forEachIndexed { index, event -> println("${index + 1}.
        ${event.eventName} - Date: ${event.eventDate}, Attendees:
        ${event.attendeeCount}")
    }
    }
    }
3 -> {
    println("\nExiting Event Manager!")
    running = false
    }
    else -> {
        println("\nInvalid choice. Please enter a valid option.")
    }
    }
    } scanner.close()
    }

```

Output: Welcome to the Event Manager!

Select an action:

1. Create a new event
2. View all events
3. Exit

Enter your choice:

1

Enter event details:

Event Name: Birthday Party Event

Date : 23-07-2024

Attendee Count: 4

Event created successfully!