Aim: Installation and basic of Android Studio & Flutter with implementation of "Hello World".

Theory:

Q1. What is Mobile App Development?

It is the process of creating software applications that run on mobile devices. Developer's design, code, test, and deploy apps for platforms like iOS and Android, ensuring functionality and user engagement.

Q2. What is different type of mobile app?

- Native Apps: Built for a specific platform (e.g., Swift for iOS, Kotlin for Android).
- Web Apps: Accessed via browsers, not installed.
- Hybrid Apps: Combine web and native technologies (e.g., using Cordova).

Q3. What is Android Studio and its Feature?

Android Studio is the official IDE for Android development. Key features include:

- Intelligent code editor
- Visual layout editor
- APK analyser
- Built-in emulator
- Fast build tools

Q3. What is AVD?

AVD (Android Virtual Device) is an emulator that mimics real Android devices. It allows developers to test apps on various virtual screen sizes, orientations, and Android versions without physical hardware.

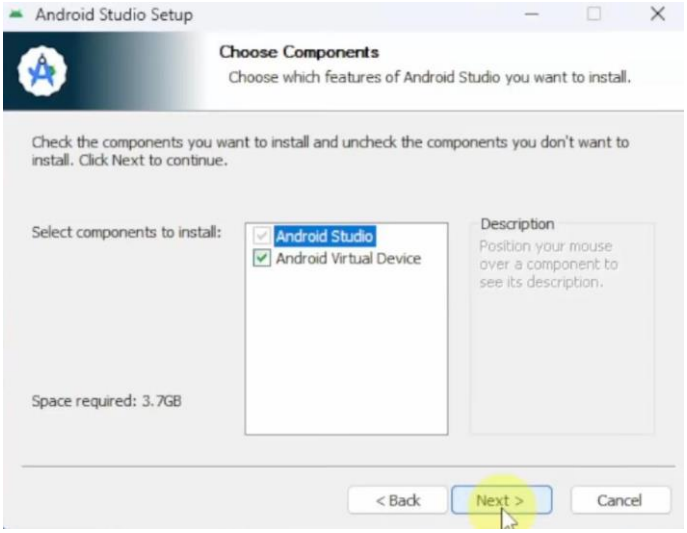
Q5. What is Flutter and its Feature?

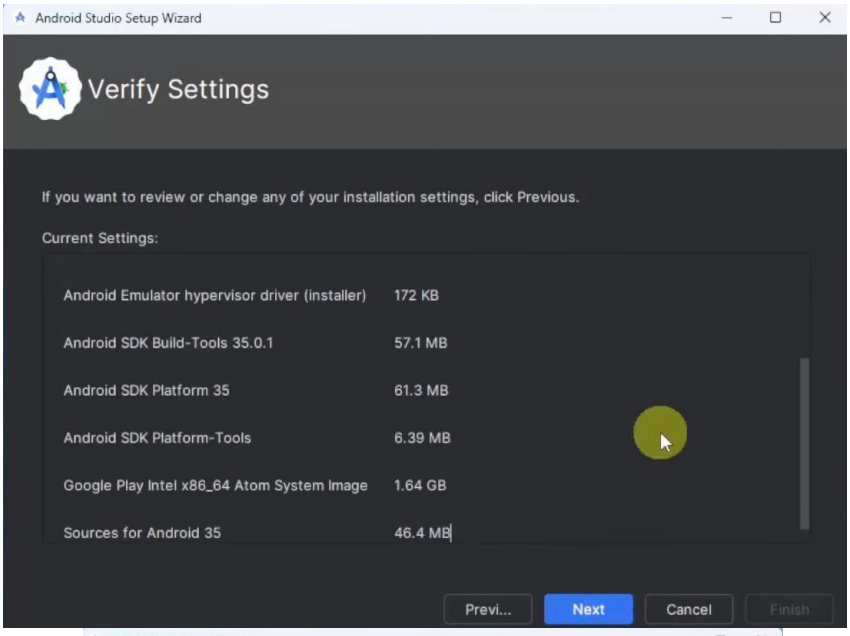
Flutter is Google's UI toolkit for building natively compiled apps from a single codebase. Features:

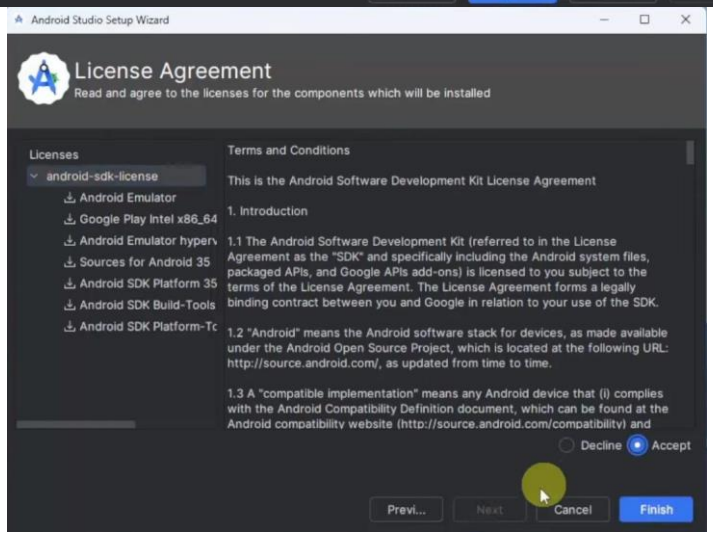
- Hot reload for quick updates
- Widget-based architecture
- Cross-platform (iOS, Android, web)
- High performance

Q6. Installation steps of Android Studio?

- Download Android Studio from the official Android Developer website.
- Run the installer and select required components such as Android SDK and Emulator.
- Follow the setup wizard and accept license agreements.
- Complete installation and launch Android Studio.
- Configure SDK Manager and download necessary SDK packages.
- Create or configure an Android Virtual Device (AVD) for app testing.


Android Studio Setup
Choose Components
Choose which features of Android Studio you want to install.
Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.
Select components to install:
Android Studio
Android Virtual Device
Description
Position your mouse over a component to see its description.
Space required: 3.7GB
< Back
Next >
Cancel


Android Studio Setup Wizard
Verify Settings
If you want to review or change any of your installation settings, click Previous.
Current Settings:
Android Emulator hypervisor driver (installer) 172 KB
Android SDK Build-Tools 35.0.1 57.1 MB
Android SDK Platform 35 61.3 MB
Android SDK Platform-Tools 6.39 MB
Google Play Intel x86_64 Atom System Image 1.64 GB
Sources for Android 35 46.4 MB
Previ...
Next
Cancel
Finish


Android Studio Setup Wizard
License Agreement
Read and agree to the licenses for the components which will be installed
Licenses
android-sdk-license
Android Emulator
Google Play Intel x86_64
Android Emulator hyperv
Sources for Android 35
Android SDK Platform 35
Android SDK Build-Tools
Android SDK Platform-Tc
Terms and Conditions
This is the Android Software Development Kit License Agreement
1. Introduction
1.1 The Android Software Development Kit (referred to in the License Agreement as the "SDK" and specifically including the Android system files, packaged APIs, and Google APIs add-ons) is licensed to you subject to the terms of the License Agreement. The License Agreement forms a legally binding contract between you and Google in relation to your use of the SDK.
1.2 "Android" means the Android software stack for devices, as made available under the Android Open Source Project, which is located at the following URL: http://source.android.com/, as updated from time to time.
1.3 A "compatible implementation" means any Android device that (i) complies with the Android Compatibility Definition document, which can be found at the Android compatibility website (http://source.android.com/compatibility) and
Decline
Accept
Previ...
Next
Cancel
Finish

Q7. Implementation of “Hello World” with output?

MainActivity.kt:

```
class MainActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()

        setContent {
            HelloWorldTheme {
                Scaffold(
                    modifier = Modifier.fillMaxSize()
                ) { innerPadding ->
                    Greeting(
                        name = "World",
                        modifier = Modifier.padding(innerPadding)
                    )
                }
            }
        }
    }
}

@Composable
fun Greeting(
    name: String,
    modifier: Modifier = Modifier
) {
    Text(
        text = "Hello $name!",
        fontSize = 32.sp,
        fontWeight = FontWeight.Bold,
        modifier = modifier
            .fillMaxSize()
            .wrapContentSize(Alignment.Center)
    )
}
```

Conclusion:

This experiment helped in understanding the basics of mobile app development, Android Studio, Flutter, and emulator usage. Implementing the “Hello World” application provided hands-on experience with the development environment and laid a strong foundation for building more advanced mobile applications in future experiments.