

# Tema\_01

## Tutorial Setup Complet - Android Studio, Java și FTC

---

### 1. Instalare Android Studio

#### Pas 1.1: Download Android Studio

##### 1. Accesează site-ul oficial:

- Mergi la [developer.android.com/studio](https://developer.android.com/studio)
- Click pe butonul verde "Download Android Studio"

##### 2. Acceptă Terms and Conditions:

- Bifează căsuța "I have read and agree..."
- Click "Download Android Studio"

##### 3. Salvează fișierul:

- Windows: android-studio-xxxx.x.x.x-windows.exe (aprox. 1 GB)
- Mac: android-studio-xxxx.x.x.x-mac.dmg
- Linux: android-studio-xxxx.x.x.x-linux.tar.gz

#### Pas 1.2: Instalare

##### Windows:

1. Rulează fișierul .exe descărcat
2. Click "Next" în wizard
3. Selectează componente (lasă toate bifate)
4. Alege locația de instalare (recomand default: C:\Program Files\Android\Android Studio)
5. Click "Install"
6. Așteaptă instalarea (5-10 minute)
7. Click "Finish" și bifează "Start Android Studio"

##### Mac:

1. Deschide fișierul .dmg
2. Trage Android Studio în folder-ul Applications
3. Deschide Applications și rulează Android Studio

4. Dacă apare warning de securitate, mergi la System Preferences > Security & Privacy și click "Open Anyway"

## Pas 1.3: First Run Setup

### 1. Setup Wizard pornește automat:

- Click "**Next**" pe ecranul de welcome

### 2. Install Type:

- Selectează "**Standard**" (recomandat)
- Click "**Next**"

### 3. Select UI Theme:

- Alege "**Darcula**" (dark theme) sau "**Light**" (după preferință)
- Click "**Next**"

### 4. Verify Settings:

- Verifică setările (ar trebui să includă Android SDK, Android Virtual Device)
- Click "**Finish**"

### 5. Downloading Components:

- Așteaptă descărcarea componentelor (10-20 minute, ~2-3 GB)
- **NU închide fereastra până nu se termină!**

### 6. Finish:

- Click "**Finish**" când totul e gata

## Pas 1.4: Verificare Instalare

### 1. Pornește Android Studio

### 2. Ar trebui să vezi ecranul de welcome cu opțiuni:

- New Project
- Open
- Get from VCS

 **Succes!** Android Studio e instalat!

---

## 2. Instalare Java JDK

### De ce ai nevoie de Java?

Android Studio vine cu Java inclus, dar pentru FTC și development general, trebuie să instalezi **Java Development Kit (JDK)** separat.

## Pas 2.1: Verificare Versiune Java Existentă

Deschide **Command Prompt** (Windows) sau **Terminal** (Mac/Linux) și scrie:

```
java -version
```

Dacă vezi ceva gen:

```
java version "17.0.x" ...
```

**Java e deja instalat!** Poți sări la [Pas 3](#).

Dacă vezi eroare, continuă cu instalarea.

## Pas 2.2: Download Java JDK

### Opțiunea 1: Oracle JDK (Recomandat pentru FTC)

1. Mergi la [oracle.com/java/technologies/downloads](https://oracle.com/java/technologies/downloads)
2. Selectează **Java 17 (LTS - Long Term Support)**
3. Alege versiunea pentru sistemul tău:
  - **Windows:** x64 Installer (.exe)
  - **Mac:** macOS Installer (.dmg) - pentru Apple Silicon sau Intel
  - **Linux:** x64 Debian Package (.deb) sau x64 RPM Package (.rpm)

### Opțiunea 2: OpenJDK (Gratuit, Open Source)

1. Mergi la [adoptium.net](https://adoptium.net)
2. Selectează **Temurin 17 (LTS)**
3. Download pentru sistemul tău

## Pas 2.3: Instalare JDK

### Windows:

1. Rulează fișierul .exe
2. Click "Next" în wizard
3. Notează path-ul de instalare (de obicei: C:\Program Files\Java\jdk-17)
4. Click "Next" și așteaptă instalarea
5. Click "Close"

### Mac:

1. Deschide fișierul .dmg
2. Dublu-click pe installer (.pkg)

3. Urmează instrucțiunile
4. Instalarea se face automat în /Library/Java/JavaVirtualMachines/

## Pas 2.4: Configurare Variabile de Mediu (IMPORTANT!)

### Windows:

#### 1. Deschide System Properties:

- Click dreapta pe "This PC" / "My Computer"
- Click "Properties"
- Click "Advanced system settings"
- Click "Environment Variables"

#### 2. Adaugă JAVA\_HOME:

- În secțiunea "System variables", click "New"
- Variable name: JAVA\_HOME
- Variable value: C:\Program Files\Java\jdk-17 (sau path-ul tău)
- Click "OK"

#### 3. Editează PATH:

- În "System variables", găsește variabila Path
- Click "Edit"
- Click "New"
- Adaugă: %JAVA\_HOME%\bin
- Click "OK" pe toate ferestrele

#### 4. Verificare:

- Deschide un Command Prompt NOU
- Scrie: java -version
- Ar trebui să vezi versiunea instalată

### Mac/Linux:

Adaugă în fișierul ~/.bash\_profile sau ~/.zshrc :

```
export JAVA_HOME=$(/usr/libexec/java_home -v 17)
export PATH=$JAVA_HOME/bin:$PATH
```

Apoi rulează:

```
source ~/.bash_profile # sau source ~/.zshrc
java -version
```

 **Succes!** Java e instalat și configurat!

### 3. Setup FTC SDK (git nu este necesar sa instalati, aveti github desktop)

#### Pas 3.1: Instalare Git

**Windows:**

1. Download Git de la [git-scm.com](http://git-scm.com)
2. Rulează installer-ul
3. Lasă toate setările default
4. Click "Next" până la final

**Mac:** Git vine pre-instalat pe Mac. Verifică cu:

```
git --version
```

**Linux:**

```
sudo apt-get install git    # Ubuntu/Debian  
sudo yum install git        # Fedora/CentOS
```

#### Pas 3.2: Clone FTC SDK

##### Opțiunea 1: Via Android Studio (Recomandat)

1. Deschide Android Studio
2. Click "**Get from VCS**" (Version Control System)
3. În câmpul URL, pune:

```
https://github.com/FIRST-Tech-Challenge/FtcRobotController.git
```

4. Alege locația unde vrei să salvezi proiectul
5. Click "**Clone**"
6. Așteaptă descărcarea (2-5 minute)

##### Opțiunea 2: Via Command Line

```
cd ~/Documents # sau folder-ul dorit  
git clone https://github.com/FIRST-Tech-Challenge/FtcRobotController.git  
cd FtcRobotController
```

Apoi deschide proiectul în Android Studio:

- File → Open
- Selectează folder-ul FtcRobotController

## Pas 3.3: Gradle Sync (IMPORTANT!)

După deschiderea proiectului:

### 1. Așteaptă Gradle Sync automat:

- Android Studio va începe automat să sincronizeze
- Vei vedea în bara de jos: "Gradle Sync in progress..."
- **Prima dată durează 10-20 minute!**
- NU Închide Android Studio!

### 2. Dacă Gradle Sync nu pornește automat:

- Click pe elefantul Gradle din toolbar
- Sau: File → Sync Project with Gradle Files

### 3. Posibile Erori:

- **"SDK location not found":**
  - File → Project Structure → SDK Location
  - Setează Android SDK path (de obicei: C:\Users\[nume]\AppData\Local\Android\Sdk )
- **"Plugin with id 'com.android.application' not found":**
  - Verifică conexiunea la internet
  - Încearcă din nou: File → Sync Project with Gradle Files

### 4. Verificare Succes:

- Când Gradle Sync se termină, vei vedea: "Gradle sync finished in Xs"
- Fără erori roșii în "Build" tab

 **Succes!** FTC SDK e instalat și configurat!

## Pas 3.4: Structura Proiectului FTC

```
FtcRobotController/
├── FtcRobotController/           ← Codul core FTC (NU modifica!)
├── TeamCode/                   ← AICI scrieți codul vostru!
│   └── src/main/java/org/firstinspires/ftc/teamcode/
│       ├── BasicOpMode_Iterative.java (exemplu)
│       ├── BasicOpMode_Linear.java (exemplu)
│       └── [aici creați fișierele voastre]
└── build.gradle                 ← Configurare build
└── gradle/                      ← Gradle wrapper
└── gradlew / gradlew.bat       ← Scripts Gradle
```

**IMPORTANT:** Tot codul vostru merge în  
TeamCode/src/main/java/org/firstinspires/ftc/teamcode/

# 4. Primul Program Java

Înainte de a programa robotul, să învățăm bazele Java cu un program simplu.

## Pas 4.1: Creează un Proiect Java Simplu

### 1. Creează un fișier Java în TeamCode:

- Click dreapta pe teamcode folder
- New → Java Class
- Nume: HelloWorld

### 2. Scrie primul program:

```
package org.firstinspires.ftc.teamcode;

public class HelloWorld {
    public static void main(String[] args) {
        // Afisează mesaj în consolă
        System.out.println("Hello, FTC!");
        System.out.println("Acesta e primul meu program Java!");

        // Operații matematice simple
        int a = 10;
        int b = 5;
        int suma = a + b;

        System.out.println("Suma dintre " + a + " și " + b + " este: " +
suma);

        // Declarații și operatori
        double putere = 0.75;
        String numeEchipa = "The Emperor";

        System.out.println("Echipa: " + numeEchipa);
        System.out.println("Putere motor: " + putere);
    }
}
```

## Pas 4.2: Rulează Programul

1. Click dreapta pe fișier
2. Run 'HelloWorld.main()'
3. Vezi output-ul în "Run" tab jos:

```
Hello, FTC!
Acesta e primul meu program Java!
Suma dintre 10 și 5 este: 15
```

Echipă: The Emperor  
Putere motor: 0.75

**Succes!** Ai scris și rulat primul program Java!

## Exercițiu 4.1: Calculator Simplu

Creează un program `Calculator.java` care:

- Declară 2 numere
- Calculează suma, diferența, produsul și împărțirea
- Afisează rezultatele

**Schelet:**

```
package org.firstinspires.ftc.teamcode;

public class Calculator {
    public static void main(String[] args) {
        // TODO: Declară variabilele
        int numar1 = 20;
        int numar2 = 4;

        // TODO: Calculează operațiile

        // TODO: Afisează rezultatele
    }
}
```

## 5. Primul OpMode FTC (doar de citit)

Acum să creăm primul OpMode care poate rula pe robot!

### Pas 5.1: Creează OpMode-ul

#### 1. Creează fișier nou:

- Click dreapta pe `teamcode`
- New → Java Class
- Nume: `PrimulMeuOpMode`

#### 2. Scrie codul:

```
package org.firstinspires.ftc.teamcode;

import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
```

```

import com.qualcomm.robotcore.eventloop.opmode.TeleOp;

@TeleOp(name="Primul Meu OpMode", group="Tutorial")
public class PrimulMeuOpMode extends LinearOpMode {

    @Override
    public void runOpMode() {
        // Inițializare
        telemetry.addData("Status", "Inițializat");
        telemetry.addData("Echipa", "The Emperor");
        telemetry.update();

        // Așteaptă butonul START
        waitForStart();

        // Loop principal - rulează după START
        while (opModeIsActive()) {
            // Afisează informații
            telemetry.addData("Status", "Rulează");
            telemetry.addData("Timp", getRuntime());
            telemetry.update();

            // Pauză mică pentru eficiență
            sleep(50);
        }
    }
}

```

## Pas 5.2: Build și Deploy

### 1. Conectează Robot Controller Phone:

- Conectează telefonul via USB
- Activează USB Debugging pe telefon

### 2. Build:

- Click pe iconița  (Build) din toolbar
- Sau: Build → Make Project
- Așteaptă build-ul să se termine

### 3. Deploy:

- Click pe iconița  (Run) din toolbar
- Selectează device-ul (telefonul)
- Așteaptă instalarea (1-2 minute)

### 4. Testează:

- Deschide Driver Station app
- Selectează "Primul Meu OpMode" din listă
- Click INIT

- Click START
- Vezi telemetry pe ecran!

**Succes!** Ai creat primul OpMode FTC!

## Exercițiu 5.1: OpMode cu Gamepad

Modifică OpMode-ul să afișeze și informații despre gamepad:

```
while (opModeIsActive()) {
    telemetry.addData("Status", "Rulează");
    telemetry.addData("Timp", getRuntime());

    // TODO: Adaugă afișare pentru:
    // - gamepad1.left_stick_y
    // - gamepad1.right_stick_x
    // - gamepad1.a (boolean)

    telemetry.update();
}
```

## 6. Git și GitHub Setup (asta doar daca vreti sa invatati sa dati commit din linie de comanda, dar ne putem descurca cu github desktop pt ca e mai usor)

### Pas 6.1: Configurare Git

Deschide Command Prompt / Terminal și configurează Git:

```
git config --global user.name "Numele Tău"
git config --global user.email "email@tau.com"
```

Verifică configurarea:

```
git config --list
```

### Pas 6.2: Creează Cont GitHub

1. Mergi la [github.com](https://github.com)
2. Click "Sign up"
3. Completează:
  - Username (ex: NumeTau\_FTC )

- Email
  - Parolă
4. Verifică email-ul
5.  Cont creat!

## Pas 6.3: Autentificare GitHub în Android Studio

1. File → Settings (Windows) sau Android Studio → Preferences (Mac)
2. Version Control → GitHub
3. Click "Add account"
4. Selectează "Log In via GitHub"
5. Se deschide browser-ul → Login cu contul GitHub
6. Autorizează Android Studio
7.  Conectat!

## Pas 6.4: Comenzi Git Esențiale

```
# Verifică status
git status

# Adaugă fișiere
git add .

# Commit
git commit -m "Mesajul commit-ului"

# Push pe GitHub
git push

# Pull (descarcă modificări)
git pull

# Clonează repository
git clone https://github.com/username/repo.git
```

---

## 7. Teme pentru Acasă

### Tema 1: Creează Repository Personal (OBLIGATORIU)

**Obiectiv:** Fiecare elev își creează propriul repository GitHub pentru exerciții.

**Pași:**

1. Creează repository nou pe GitHub:

- Mergi la [github.com](https://github.com)
- Click pe butonul verde "New" sau "+" → "New repository"
- Completează:
  - **Repository name:** Teme\_[NumeleTau] (ex: Teme\_Andrei`)
  - **Description:** "Exerciții și proiecte pentru cursul FTC"
  - **Vizibilitate:** Public (să putem vedea toți)
  - Bifează "Add a README file"
  - Lasă restul nebifare
- Click "Create repository"

## 2. Clone repository-ul local:

- Copiază URL-ul repository-ului (butonul verde "Code")
- În Android Studio:
  - File → New → Project from Version Control
  - Paste URL-ul
  - Alege locația
  - Click "Clone"

## 3. Creează structura de foldere:

```

Teme_[NumeleTau]/
├── README.md
├── Curs1_Fundamentals/
├── Curs2_OOP/
├── Curs3_Subsystems/
└── Proiect_Final/

```

## 4. Editează README.md:

```

# FTC Curs - [Numele Tău]

## Despre Mine
- **Nume:** [Numele tău]
- **Echipă FTC:** The Emperor (#124)
- **Rol:** Programator

## Progres Curs
- [ ] Curs 1: Fundamentals
- [ ] Curs 2: OOP
- [ ] Curs 3: Subsystems
- [ ] Proiect Final

## Proiecte
- [Link către primul OpMode]
- [Link către primul subsistem]

```

## 5. Push primul commit:

```
git add .
git commit -m "Setup inițial - structură repository"
git push
```

#### 6. Verificare:

- Accesează repository-ul pe GitHub
  - Verifică că README.md apare formatat
  - Repository creat cu succes!
- 

## Tema 2: Calculator Avansat (Optional)

**Obiectiv:** Practică sintaxa Java.

**Creează:** Curs1\_Fundamentals/CalculatorAvansat.java

**Cerințe:**

1. Citește 2 numere (hardcodate în cod)
2. Calculează și afișează:
  - Suma
  - Diferența
  - Produsul
  - Împărțirea (cu verificare pentru 0!)
  - Restul împărțirii (modulo)
  - Puterea (numărul 1 la puterea numărul 2)
3. Formatează output-ul frumos

**Exemplu output:**

```
==== CALCULATOR AVANSAT ====
Numărul 1: 10
Numărul 2: 3
```

**Rezultate:**

```
10 + 3 = 13
10 - 3 = 7
10 * 3 = 30
10 / 3 = 3.33
10 % 3 = 1
10 ^ 3 = 1000
```

# Resurse Suplimentare

## Video Tutorial

- [FTC Setup Tutorial](#) - Tutorial video complet pentru setup

## Documentație

- [Android Studio User Guide](#)
- [FTC Documentation](#)
- [Java Tutorials](#)
- [Git Documentation](#)

## Comunitate

- [FTC Discord](#)
  - [r/FTC Reddit](#)
  - [FTC Forum](#)
- 

## Troubleshooting Comun

### Problemă: "SDK location not found"

#### Soluție:

1. File → Project Structure
2. SDK Location
3. Setează path-ul: C:\Users\[nume]\AppData\Local\Android\Sdk (Windows)

### Problemă: Gradle Sync eșuează

#### Soluție:

1. Verifică conexiunea la internet
2. File → Invalidate Caches / Restart
3. Încearcă din nou: File → Sync Project with Gradle Files

### Problemă: "Unable to resolve dependency"

#### Soluție:

1. Verifică fișierul build.gradle
2. Asigură-te că ai ultima versiune de FTC SDK
3. Încearcă: Build → Clean Project, apoi Build → Rebuild Project

## **Problemă: Git push eșuează**

**Soluție:**

1. Verifică autentificarea: `git config --list`
2. Încearcă: `git pull` înainte de `git push`
3. Dacă persistă, folosește GitHub Desktop

## **Problemă: Java not found**

**Soluție:**

1. Verifică instalarea: `java -version`
2. Verifică variabilele de mediu: `echo %JAVA_HOME%` (Windows)
3. Reinstalează JDK dacă e necesar