



دائماً معاك



Dart OOP

Task 1.1 : Create a Student Class

Create a Student class with name, age, and favoriteLanguage. Add a method printDetails() that prints the student info.

- Hint: Use a constructor and simple print statement inside the method.
- Output : Student: Amro, Age: 27, Favorite Language: ReactJS

Task 1.2 : Create a Method to Update Favorite Language

Extend the Student class to include a method updateFavoriteLanguage() that changes the student's favorite language.

- Hint: Use a setter-like method to update the **favoriteLanguage**.
- Output: Updated favorite language to Dart

Task 1.3 : Add a Method to Calculate Student's Year of Birth

Add a method calculateYearOfBirth() to the Student class that calculates the student's year of birth based on their age.

- Hint: Use **DateTime.now().year** to get the current year and subtract the age.
- Output: Amro's year of birth: 1997

Task 1.4 : Create a Class to Track Multiple Students

Create a new class StudentManager that can store a list of Student objects. Add methods to add a new student and print all students' details.

- Hint: Use a List<Student> to store students and a loop to print all details.
- Output: Student: Amro, Age: 27, Favorite Language: ReactJS / Student: Lina, Age: 22, Favorite Language: Dart

Task 1.5 : Use a Named Constructor for Guest Student

- Create a named constructor called Student.guest() that initializes a student with default values: name = "Guest" / age = 0 / favoriteLanguage = "Unknown"
- Hint: Use a named constructor in Dart to provide default guest information.
- Output: Student: Guest, Age: 0, Favorite Language: Unknown

Coach Amro





دائماً معاك



Dart OOP

Task 1.6 : Encapsulate Favorite Language with Getter and Setter

Make the favoriteLanguage property private and provide a getter and a setter:

- The setter should only update the language if the value is not empty.
- Hint: Use `_favoriteLanguage` as a private field, and define a getter and setter to access and modify it.
- Output : Current favorite language: ReactJS / Updated favorite language to Dart

Task 1.7 : Encapsulate and Validate Age

Make the age field private and use a getter and setter to control access:

- The setter should only accept positive values. Negative values should be ignored.
- Hint: Use `_age` as a private variable and validate the value in the setter.
- Output: Student age: 25 / Trying to set age to -5... / Age remains: 25

