

## Developing Object-Oriented Programming

**Course/Duty Area:** Programming/Developing Object-Oriented Programming

**Scenario:**

### 1. Introduction to Object-Oriented Programming (OOP)

**Objective:** Students will understand the fundamental concepts of OOP, including classes and objects.

**Key Points:**

- Definition of OOP
- Benefits of OOP
- Overview of classes and objects

**Assessment:** Class discussion and quick quiz on key terms (e.g., class, object, instance).

### 2. Principles of OOP

**Objective:** Students will explore the four main principles of OOP: encapsulation, inheritance, polymorphism, and abstraction.

**Key Points:**

- Explanation of each principle
- Examples in real-world scenarios
- Discussion on how these principles enhance programming

**Assessment:** Group/individual activity where students create a visual representation of the four principles.

### 3. Steps in OOP Development

**Objective:** Students will learn the steps involved in developing an OOP program, including planning, designing, coding, and testing.

**Key Points:**

- Breakdown of the development process
- Importance of each step in successful programming
- Discussion on common pitfalls

**Assessment:** Create a flowchart that outlines the steps of OOP development

#### 4. **Creating User-Defined Classes**

**Objective:** Students will create their own user-defined classes in a programming language of choice (e.g., Python, Java).

**Key Points:**

- Syntax for defining classes
- Attributes and methods
- Creating objects from classes

**Assessment:** Submit a simple program that includes at least one user-defined class with attributes and methods.

#### 5. **Project Day - Building an OOP Application**

**Objective:** Students will apply their knowledge of OOP to develop a small application that utilizes user-defined classes.

**Key Points:**

- Review of class definitions and principles
- Work on projects
- Peer feedback

**Assessment:** Presentation of the final project showcasing the application and explaining the classes created.

**Big Question:** Do you understand OOP, and can you write a user-defined program in the language of your choice?

**Focused Questions:**

- Did you design and create methods and properties for each class?
- Does the program use the class parameter of the create function statement?
- Does the program run successfully?

**Student Project or Outcome:** Students will create a user-defined class program. Students will understand the components that go into creating this type of program.

**Project-Based Assessment:** Presentation of the final project showcasing the application and explaining the classes created.

*Scenario submitted by Elisa Bradford, Blue Ridge Technical Center, Warren County Public Schools*