📘 Understanding Class Instances in C++

🧠 What is a Class Instance?

A **class instance** (also called an **object**) is a concrete realization of a class. While a class defines the **structure** and **behavior** (via fields and methods), an instance represents a **specific entity** with its own **state**.

🔑 Key Concepts:

- A class is like a **blueprint**.
- An instance is like a **house built from that blueprint**.
- Each instance has its own **copy of the class's fields**.

🧱 Declaring and Creating Instances

🔹 Syntax

```
ClassName objectName;               // Default constructor
ClassName objectName(args);        // Parameterized constructor
```

🍭 Example

```
class Car {
  public:
    string brand;
    int year;

    void display() {
        cout << "Brand: " << brand << ", Year: " << year << endl;
    }
};

int main() {
    Car car1;               // Instance of Car
    car1.brand = "Toyota";
    car1.year = 2020;

    car1.display();         // Output: Brand: Toyota, Year: 2020

    return 0;
}
```

🚀 Instances with Constructors

```
class Student {
  private:
    string name;
    int age;

  public:
    Student(string n, int a) : name(n), age(a) {}

    void display() {
        cout << "Name: " << name << ", Age: " << age << endl;
    }
};

int main() {
    Student s1("Alice", 20);   // Instance with parameterized constructor
    Student s2("Bob", 22);

    s1.display();              // Output: Name: Alice, Age: 20
    s2.display();              // Output: Name: Bob, Age: 22

    return 0;
}
```

✅ Summary

Term	Description
Class	Blueprint for objects
Instance	A specific object created from a class
Constructor	Initializes an instance

🎯 Quiz!

Here's a short quiz on the topic: quiz

🔭 Markdown Viewer

How to view the markdown files in a browser...

- Markdown Viewer

🧠 Lecture Practices

Here are the lecture Practices...

- Day 7
- Day 8
- Day 9

🔍 Lecture Quizzes

Here are the lecture quizzes...

- Day 7
- Day 8
- Day 9

Weekly Topics

Here are the topics for the week...

- Classes
- Structs
- Fields
- Getters and Setters
- Constructors
- Instances
- Inheritance
- Polymorphism
- Pointers
- Upcasting
- Misc. Concepts
- 4 Pillars of OOP