It looks like you're asking about data types in relation to classes in programming. In object-oriented programming (OOP), a class is a blueprint for creating objects. These objects can store data and have methods to manipulate that data. Here are some key terms and concepts:

1. \*\*Fields (Attributes)\*\*: Variables that hold data for an object. They can be of various data types like `int`, `float`, `String`, or even other objects. For example:

```java

class Student {

int rollNumber;

String name;

float marks;

}

```

2. \*\*Methods\*\*: Functions defined inside a class that operate on the objects of the class. Methods can use and modify the object's fields. For example:

```java

class Student {

int rollNumber;

String name;

float marks;

void displayDetails() {

System.out.println("Roll Number: " + rollNumber);

System.out.println("Name: " + name);

System.out.println("Marks: " + marks);

}

}

```

3. \*\*Constructors\*\*: Special methods used to initialize objects. Constructors have the same name as the class and do not have a return type. For example:

```java

class Student {

int rollNumber;

String name;

float marks;

Student(int rollNumber, String name, float marks) {

this.rollNumber = rollNumber;

this.name = name;

this.marks = marks;

}

}

```

4. \*\*Access Modifiers\*\*: Keywords that determine the visibility of fields and methods. Common access modifiers include `private`, `protected`, and `public`. For example:

```java

class Student {

private int rollNumber;

private String name;

private float marks;

public void setRollNumber(int rollNumber) {

this.rollNumber = rollNumber;

}

public int getRollNumber() {

return rollNumber;

}

}

```

These components together define the data and behavior of objects created from the class. If you have any specific questions about classes or data types, feel free to ask!

It seems like you meant to ask about the "type" of a variable in a class. In object-oriented programming (OOP), variables in a class are known as attributes or fields, and they can have different types, such as primitives (int, float, etc.) or object types (instances of other classes).

Here's a quick rundown of how class type variables are used in Java:

1. \*\*Primitive Types\*\*: These are the most basic data types and include `int`, `float`, `double`, `char`, etc.

```java

class Example {

int integerField;

float floatField;

}

```

2. \*\*Object Types\*\*: These are instances of other classes. When a class includes variables that are of a class type, it means these variables can hold objects.

```java

class Student {

String name; // String is a class in Java

int rollNumber;

}

class Classroom {

Student student; // student is an object of type Student

}

```

In the above example:

- `String` is a built-in class in Java, so `name` is an object of the `String` class.

- `student` in the `Classroom` class is an object of type `Student`.

This enables you to create complex data structures and relationships between objects. If you need more specific details or examples, feel free to let me know!