Chapter 13

Overview of Variables, Data types & Methods

Sweet pan తీపి కిళ్ళీ

Goal

Sweet Pan

Sweet pan తీపి కిళ్ళీ

Variable

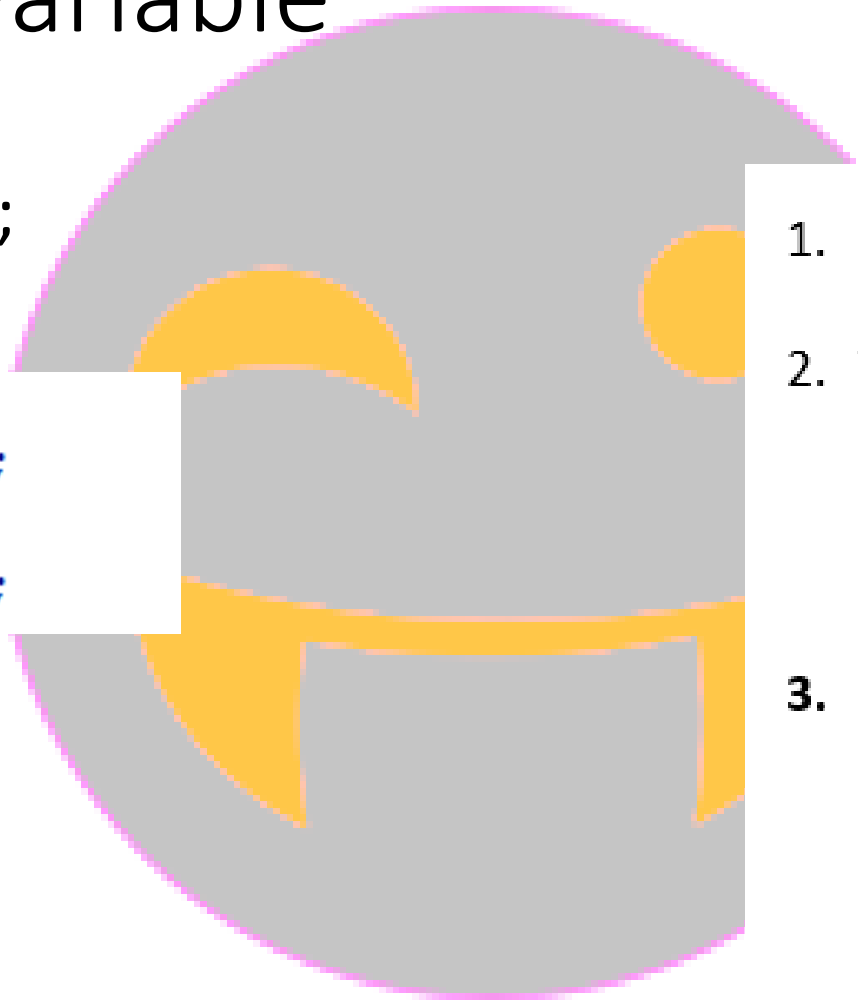
Box-1, Box-2, Box-3, Box-4, Box-5

1. A Variable **is like a container**, which **holds some data**
2. The **container(variable)** will have **specific size(300g)** associated with it, so that we can **store maximum that much data**
3. **Data**(Ingredient) inside the container(variable) **can change at any time** while making pan(executing java program)

Declaring a variable

datatype identifier;

```
String name;
String studyClass;
int rollno;
double percentage;
```



1. A Variable **is like a container**, which **holds some data**
2. The **container(variable)** will have **specific size(300g)** associated with it, so that we can **store maximum that much data**
3. **Data**(Ingredient) inside the container(variable) **can change at any time** while making pan(executing java program)

Write an application for **cash calculation**

+

+

Suitcase-1 Suitcase-2 Suitcase-3

System.out.println(Suitcase1+Suitcase2+Suitcase3);

CashProgram.java

```
class CashProgram{
    public static void main(String[] args){
        suitcase1 = 1000;
        suitcase2 = 2000;
        suitcase3 = 3000;
        System.out.println(suitcase1+suitcase2+suitcase3);
    }
}
```

Types of programming languages

- **Statically typed language**
- The **type of the values** that a variable can hold is decided during the **compile time** of the program
- We **can't assign any other types of values** once decided a type
- Ex: C, C++, **Java** etc.

- **Dynamically typed language**
- The **type of the values** that a variable can hold is decided during the **run time** of the program
- We **can assign any type of values**(Interpreter assigns variables a type at runtime based on the variables values at that time)
- Ex: Python, JavaScript, Ruby etc.

NOTE: Java is a statically typed language

Adding data types to variables(variable declaration)

- datatype identifier;
- datatype identifier = value;

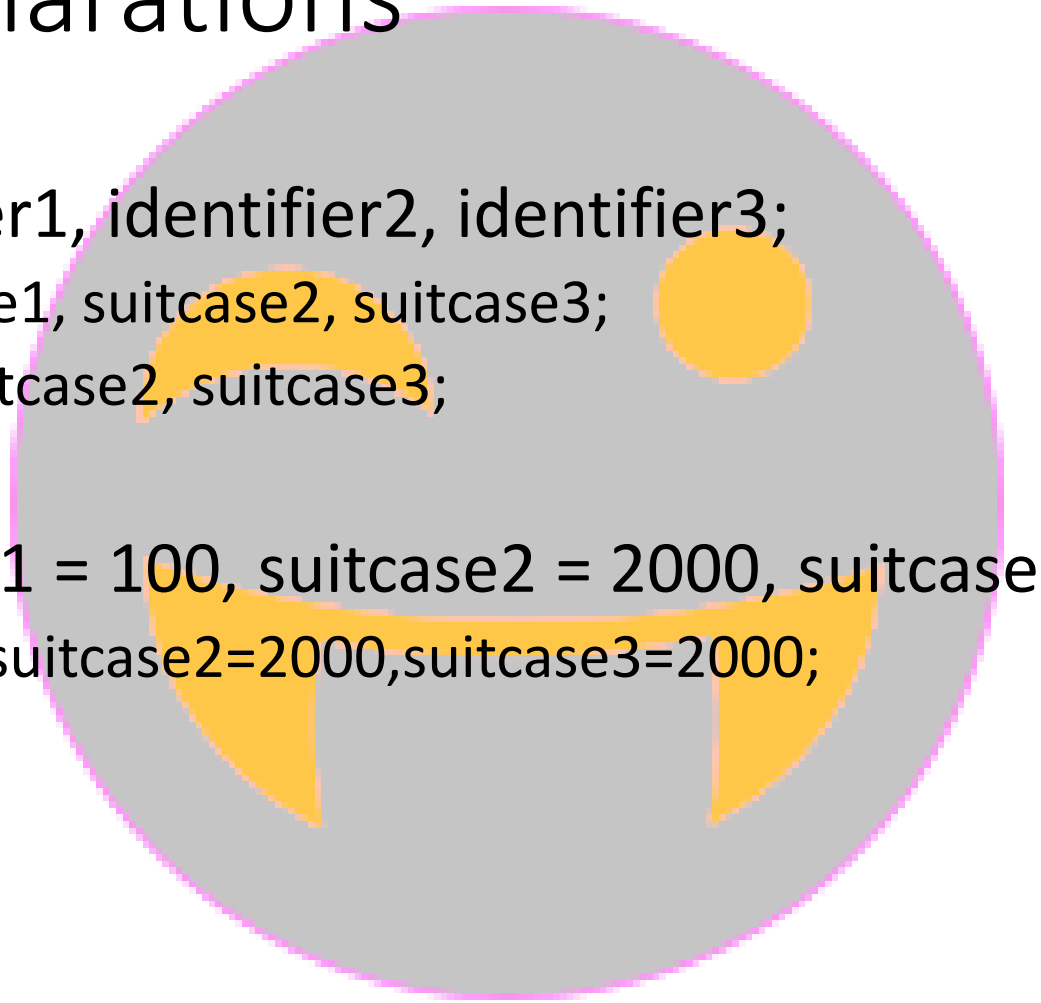
```
class CashProgram{
    public static void main(String[] args){
        int suitcase1 = 1000;
        int suitcase2 = 2000;
        int suitcase3 = 3000;
        System.out.println(suitcase1+suitcase2+suitcase3);
    }
}
```

Java is statically typed language

Possible declarations

- datatype identifier1, identifier2, identifier3;
  - datatype suitcase1, suitcase2, suitcase3;
  - int suitcase1, suitcase2, suitcase3;

- datatype suitcase1 = 100, suitcase2 = 2000, suitcase3 = 2000;
  - suitcase1=1000,suitcase2=2000,suitcase3=2000;



What is the output below program

```
class CashProgram{
    public static void main(String[] args){
        int suitcase1 = 1000;
        int suitcase2 = 2000;
        int suitcase3 = 3000;
        System.out.println(suitcase1+suitcase2+suitcase3);
        System.out.print(suitcase1);
        System.out.print(suitcase2);
        System.out.print("suitcase3");
    }
}
```

```
6000
10002000suitcase3
```


Variables Review

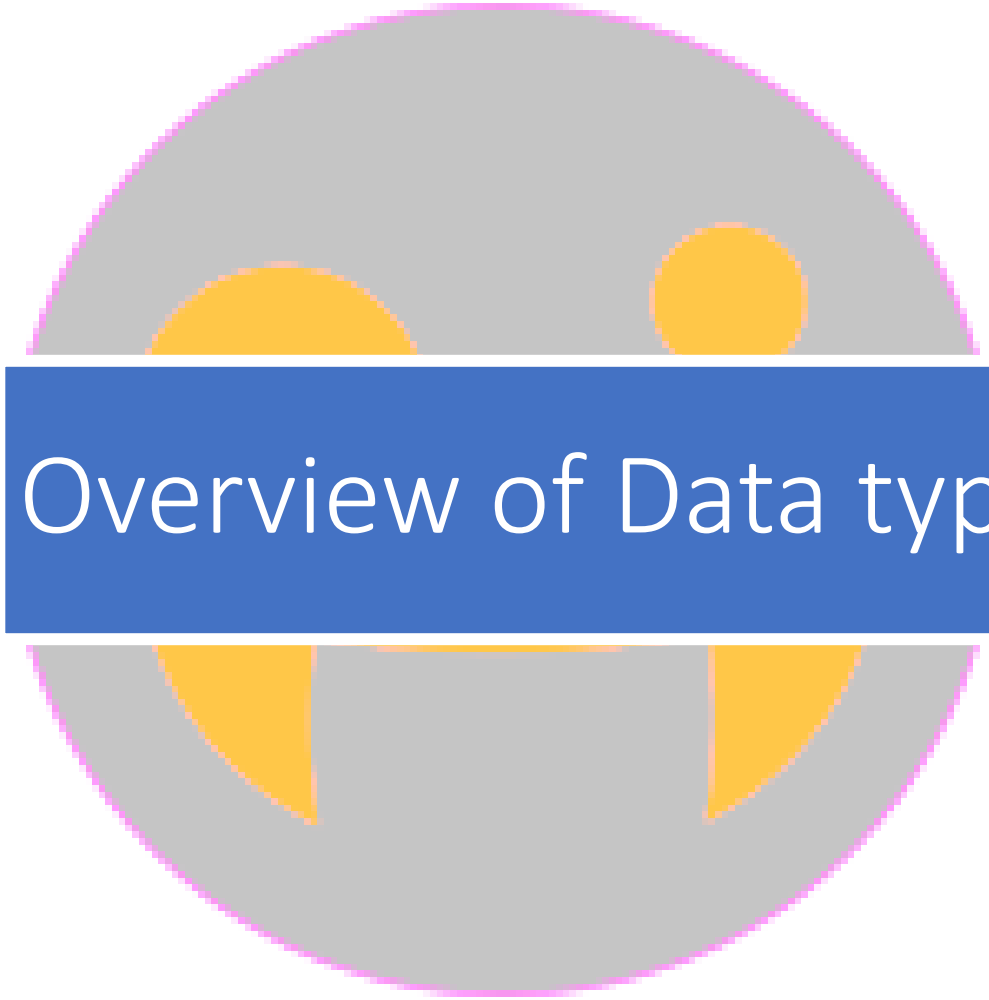
Variables

1. A Variable **is like a container**, which **holds some data**
2. The **container(variable)** will have **specific size(300g)** associated with it, so that we can **store maximum that much data**

- Variable is **used to store data.**
  - Ex: int suitcase1 = 1000
- Every **variable is assigned a data type** that describes the **type** and **quantity** of value it can hold
- Value stored in a variable can be **changed anytime** during program execution

```
class CashProgram{
    public static void main(String[] args){
        int suitcase1 = 1000;
        int suitcase2 = 2000;
        int suitcase3 = 3000;
        System.out.println(suitcase1+suitcase2+suitcase3);
        System.out.print(suitcase1);
        System.out.print(suitcase2);
        System.out.print("suitcase3");
    }
}
```

```
class CashProgram{
    public static void main(String[] args){
        int suitcase1 = 1000;
        int suitcase2 = 2000;
        int suitcase3 = 3000;
        System.out.println(suitcase1+suitcase2+suitcase3);
        System.out.print(suitcase1);
        System.out.print(suitcase2);
        System.out.print("suitcase3");
        suitcase1 = 500;
        System.out.println(suitcase1+suitcase2+suitcase3);
    }
}
```

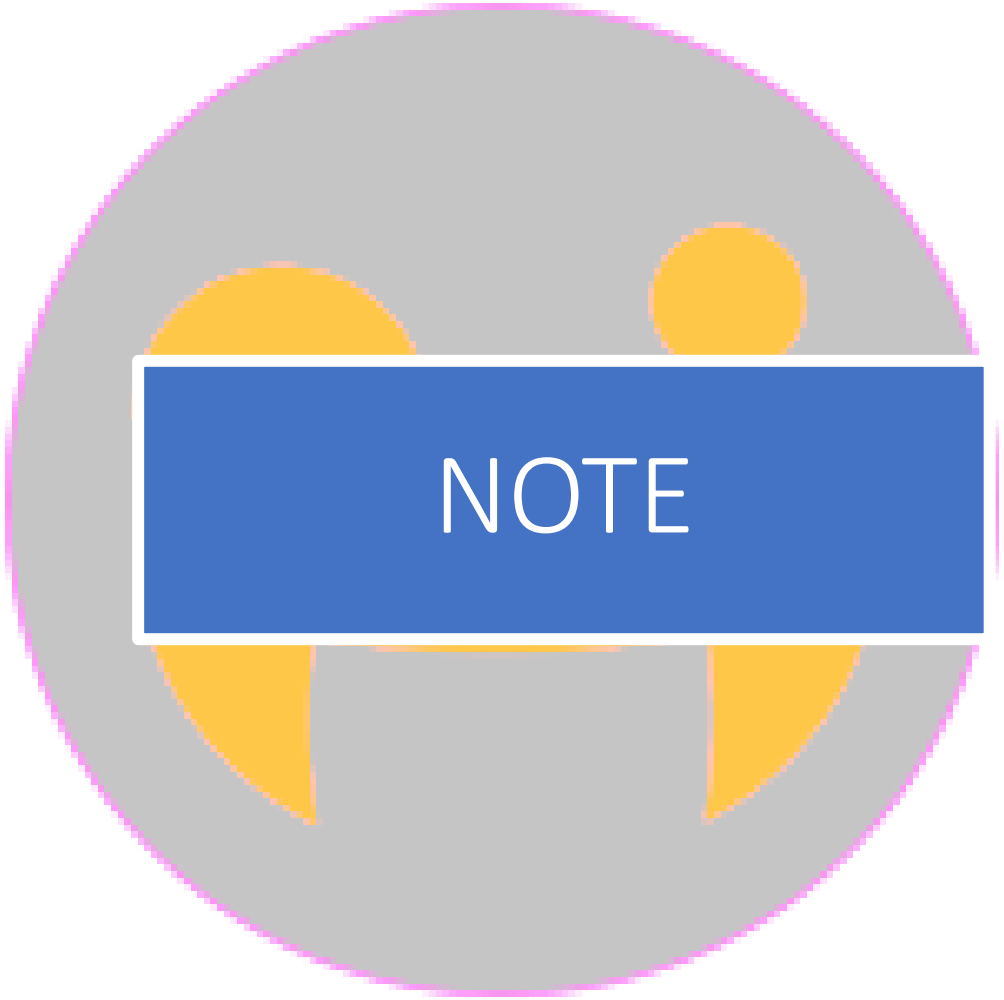


Overview of Data types

Data types

- Defines the **type of the data** that can be **stored in the variable**
- Two types:
  - **Primitive data types(8)**
    - **byte**
    - **short**
    - **int**
    - **long**
    - **double**
    - **float**
    - **boolean**
    - **char**
  - **Non primitive data types (reference type)**

```
class CashProgram{
    public static void main(String[] args){
        int suitcase1 = 1000;
        int suitcase2 = 2000;
        int suitcase3 = 3000;
        System.out.println(suitcase1+suitcase2+suitcase3);
        System.out.print(suitcase1);
        System.out.print(suitcase2);
        System.out.print("suitcase3");
    }
}
```


NOTE

suitcase1,suitcase2,suitcase3 are **local variables**

- **Local Variables**
- Variables that are **declared inside the method** are called **local variables**
- They are called local because they are **local to the method**
- NOTE:
- Local variables doesn't get a **default value**
- Local variables should be assigned with a value before used
- Reassignment is also possible
  - int suitcase1;
  - suitcase1 = 1000;
- **Just make sure that the variable has some value in it before using it**

```
class CashProgram{
    public static void main(String[] args){
        int suitcase1 = 1000;
        int suitcase2 = 2000;
        int suitcase3 = 3000;
        System.out.println(suitcase1+suitcase2+suitcase3);
        System.out.print(suitcase1);
        System.out.print(suitcase2);
        System.out.print("suitcase3");
    }
}
```

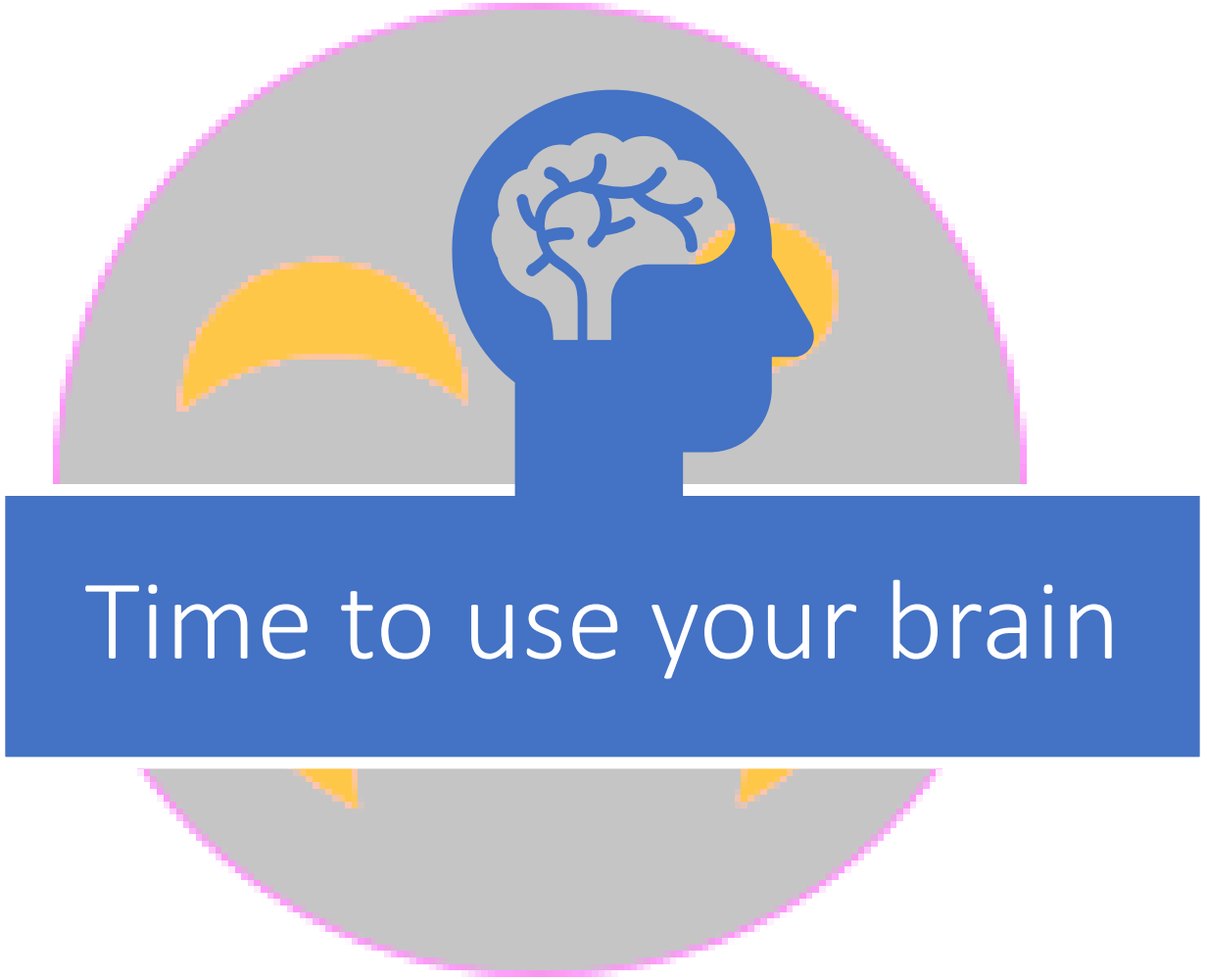
Instance variables

Within the body of the class, but outside of the method

Why are they are called instance variables will be discussed later

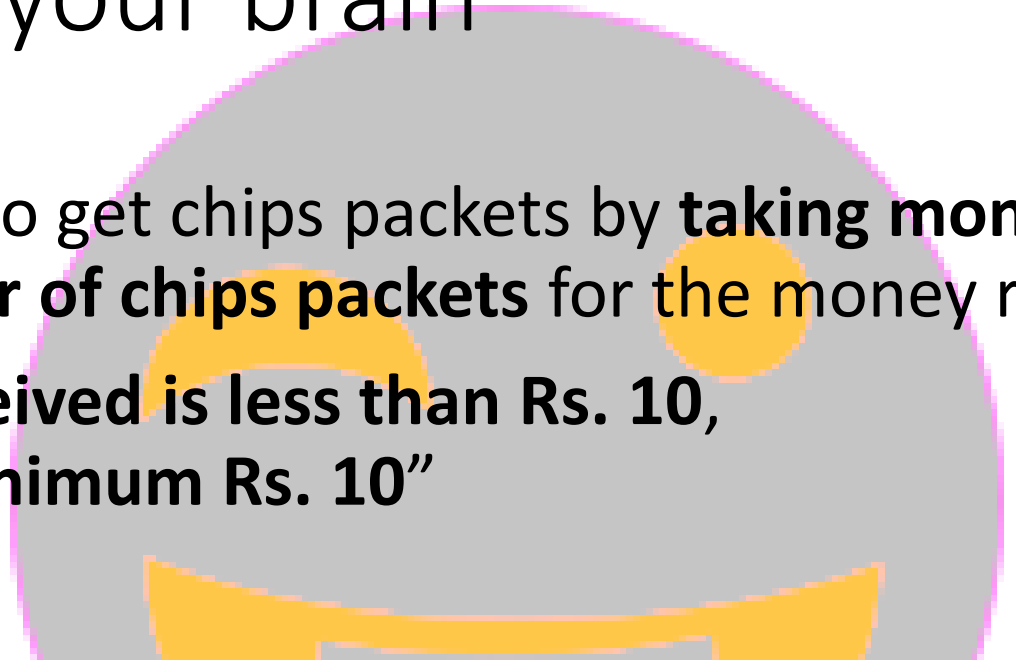
Static variables

```
class Student{
String name;
String studyclass;
int rollno;
double percentage;

void setStudyClass(String sc){

}
void setRollno(int rn){

}
void setPercentage(double percentage){

}

String getStudyClass(){
    return studyclass;
}
int getRollno(){
    return rollno;
}
double getPercentage(){
    return percentage;
}
}
```



Time to use your brain

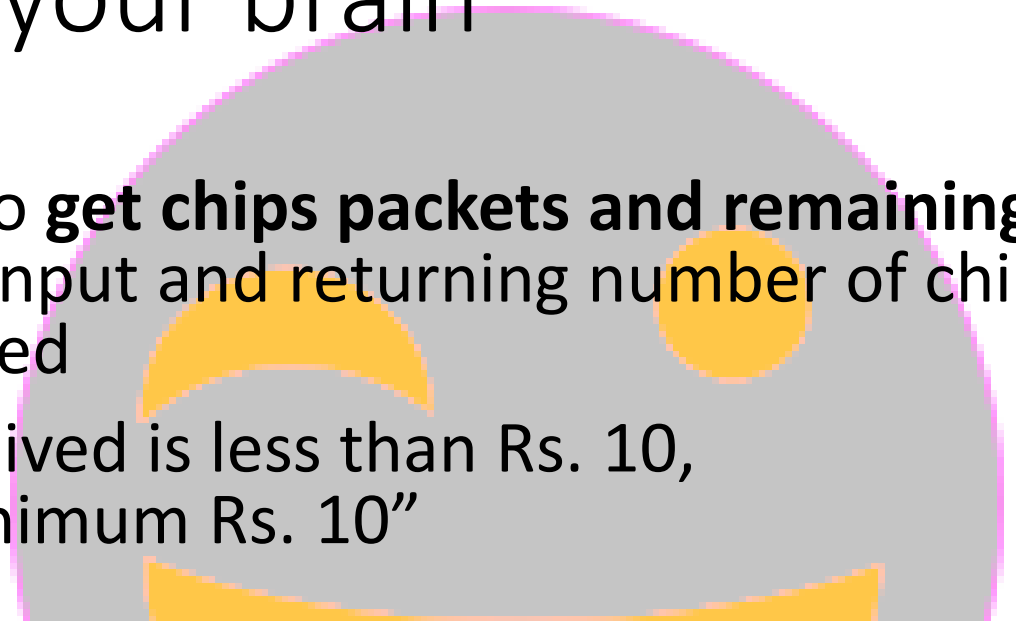
Time to use your brain



- **Write a method** to get chips packets by **taking money as input** and **returning number of chips packets** for the money received
- **If the money received is less than Rs. 10**, return "**Sorry, minimum Rs. 10**"

```
static String getChipsPackets(int cash){
    if(cash<10){
        return "Sorry, minimum Rs. 10";
    }else{
        return "e "+cash/10+" chips packets teesukondi";
    }
}
```

Time to use your brain



- Write a method to **get chips packets and remaining amount** by taking money as input and returning number of chips packets for the money received
- If the money received is less than Rs. 10, return “Sorry, minimum Rs. 10”

```
static String getChipsPackets(int cash){
    if(cash<10){
        return "Sorry, minimum Rs. 10";
    }else{
        int chipsPackets = cash/10;
        int reminingAmount = cash%10;
        String message = "e "+chipsPackets+" chips packets teesukondi.";
        if(reminingAmount!=0){
            message = message + "e Rs."+reminingAmount+" chillara teesukondi";
        }
        return message;
    }
}
```

Time to use your brain

```
static String getChipsPackets(int cash){
    if(cash<10){
        return "Sorry, minimum Rs. 10";
    }else{
        return "e "+cash/10+" chips packets teesukondi";
    }
}
```

```
static String getChipsPackets(int cash){
    if(cash<10){
        return "Sorry, minimum Rs. 10";
    }else{
        int chipsPackets = cash/10;
        int reminingAmount = cash%10;
        String message = "e "+chipsPackets+" chips packets teesukondi.";
        if(reminingAmount!=0){
            message = message + "e Rs."+reminingAmount+" chillara teesukondi";
        }
        return message;
    }
}
```

NOTE

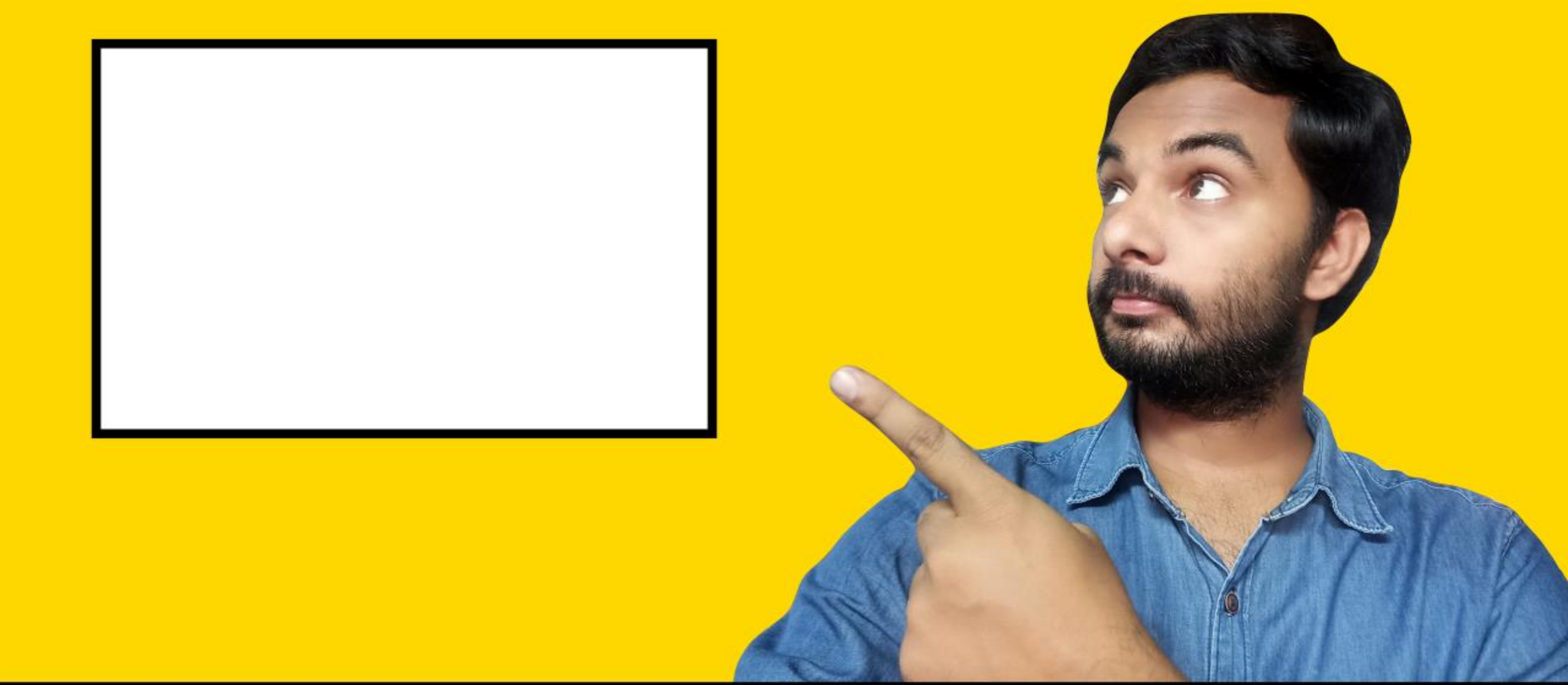
- Will discuss more about variables and data types later 😊 😊 😊

```
class CashProgram{
    public static void main(String[] args){
        int suitcase1 = 1000;
        int suitcase2 = 2000;
        int suitcase3 = 3000;
        System.out.println(suitcase1+suitcase2+suitcase3);
        System.out.print(suitcase1);
        System.out.print(suitcase2);
        System.out.print("suitcase3");
        suitcase1 = 500;
        System.out.println(suitcase1+suitcase2+suitcase3);
    }
}
```

- As the **program grows, it is difficult** to understand
- We can provide **comments(notes)** to refer it later
- Will be **useful for other developers** to understand why was that piece of code used
- **Let's talk about comments**

What next?

Comments



చిన్న బ్రేక్ చిటికలో వచ్చేస్తా