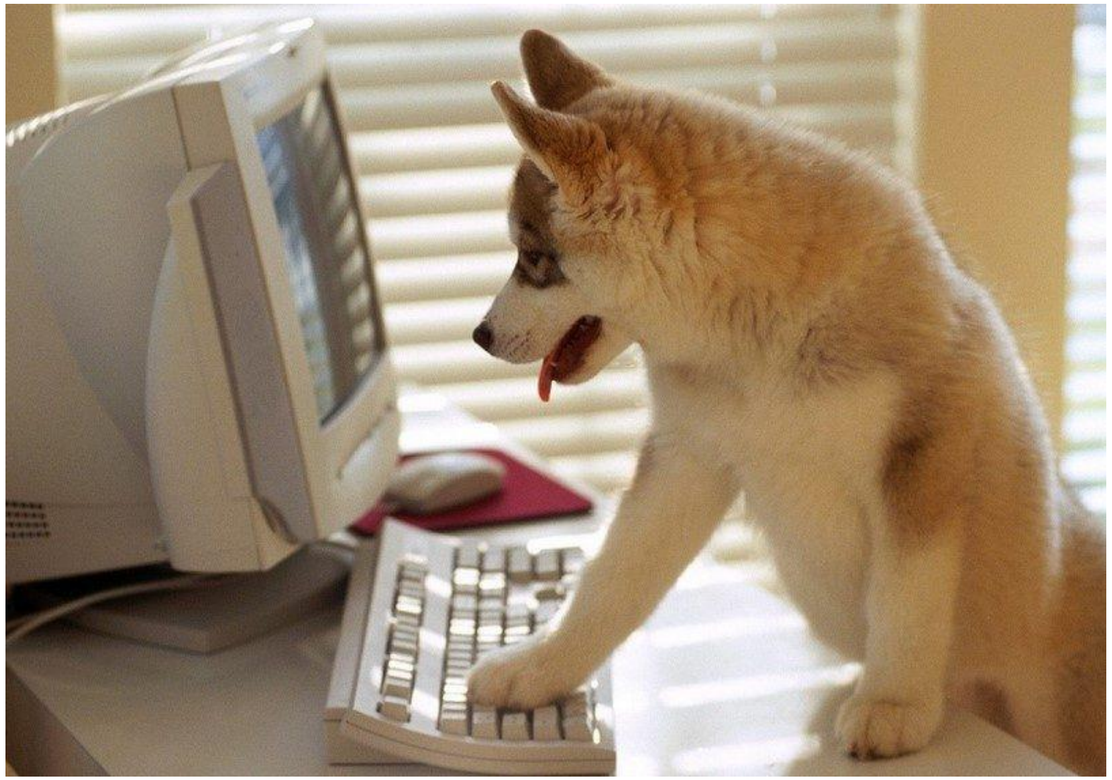
Java Programming



Organizational Stuff

23.09.: No Students :(
24.09.: Structures
25.09.: Methods
26.09.: Recursion
27.09.: Arrays & Strings

30.09.: OOP
01.10.: Generics & Linked Lists
02.10.: Exceptions & Testing
03.10.: Holiday
04.10.: GUI

Recursion

Something defined by itself

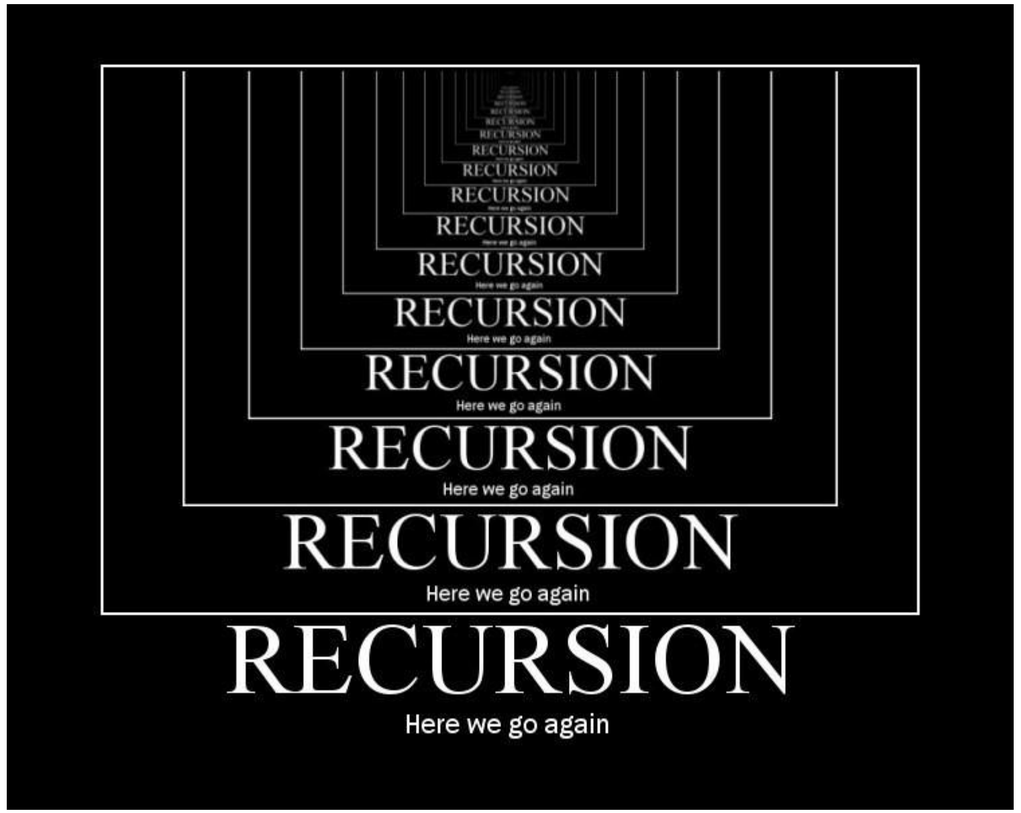


Image: https://cdn-images-1.medium.com/max/1600/1*appBwh6_RtvocVxwqpplHA.jpeg

Recursion

```
public static int rekExample(int x){
  if(x<1){
    return x;
  }
  else{
    System.out.println("Recursion is fun!");
    return rekExample(x-1);
  }
}
```

Recursion

```
public static int rekExample(int x){   //x=3
  if(x<1){
    return x;
  }
  else{
    System.out.println("Recursion is fun!");
    return rekExample(x-1);
  }
}
```

Recursion

```
public static int rekExample(int x){   //x=3
  if(x<1){
    return x;
  }
  else{
    System.out.println("Recursion is fun!");
    return rekExample(x-1);
  }
}
```

Recursion is fun!

Recursion

```
public static int rekExample(int x){   //x=3
  if(x<1){
    return x;
  }
  else{
    System.out.println("Recursion is fun!");
    return rekExample(x-1);    //x=2
  }
}
```

Recursion is fun!

Recursion

```
public static int rekExample(int x){   //x=2
  if(x<1){
    return x;
  }
  else{
    System.out.println("Recursion is fun!");
    return rekExample(x-1);    //x=2
  }
}
```

Recursion is fun!

Recursion

```
public static int rekExample(int x){   //x=2
  if(x<1){
    return x;
  }
  else{
    System.out.println("Recursion is fun!");
    return rekExample(x-1);    //x=2
  }
}
```

Recursion is fun!
Recursion is fun!

Recursion

```
public static int rekExample(int x){   //x=2
  if(x<1){
    return x;
  }
  else{
    System.out.println("Recursion is fun!");
    return rekExample(x-1);    //x=1
  }
}
```

Recursion is fun!
Recursion is fun!

Recursion

```
public static int rekExample(int x){   //x=1
  if(x<1){
    return x;
  }
  else{
    System.out.println("Recursion is fun!");
    return rekExample(x-1);    //x=1
  }
}
```

Recursion is fun!
Recursion is fun!

Recursion

```
public static int rekExample(int x){   //x=1
  if(x<1){
    return x;
  }
  else{
    System.out.println("Recursion is fun!");
    return rekExample(x-1);    //x=1
  }
}
```

Recursion is fun!
Recursion is fun!
Recursion is fun!

Recursion

```
public static int rekExample(int x){   //x=1
  if(x<1){
    return x;
  }
  else{
    System.out.println("Recursion is fun!");
    return rekExample(x-1);    //x=0
  }
}
```

Recursion is fun!
Recursion is fun!
Recursion is fun!

Recursion

```
public static int rekExample(int x){   //x=0
  if(x<1){
    return x;
  }
  else{
    System.out.println("Recursion is fun!");
    return rekExample(x-1);    //x=0
  }
}
```

Recursion is fun!
Recursion is fun!
Recursion is fun!

Recursion

```
public static int rekExample(int x){   //x=0
  if(x<1){
    return x;
  }
  else{
    System.out.println("Recursion is fun!");
    return rekExample(x-1);    //x=0
  }
}
```

Recursion is fun!
Recursion is fun!
Recursion is fun!

Recursion

```
public static int rekExample(int x){
  if(x<1){
    return x;
  }
  else{
    System.out.println("Recursion is fun!");
    return rekExample(x-1);
  }
}
```

Recursion

Today's Assignment:
https://classroom.github.com/a/U6vstOSW