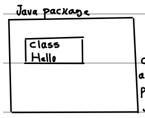
1/28/26 lecture #2)

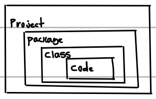


code lives in a class,
and classes live in
packages, and packages
lives in an IDE project,
more specifically (in our case)

they live in an eclipse project.

Using eclipse:

1. set up a project
2. set up a package inside the project
3. set up a class inside the package
4. write code inside the class



in eclipse, a source directory (src) is contained in your Project

src will contain your packages, by convention: packages are named in lower case, classes are in upper case.

ex code in lecture

```
package class02; //Where the class is stored/contained
public class Hello {
   public static void main(string[] args) {
      System.out.println ("Hello world");
   }
```

the code contained in your class

Static functions in Java (analogous to functions in c++)

e.g. let's write a function to square an integer.

inside the main function:

```
int x=square(5); //now let's set it up
System.out.println(X);
```

setting up a static function:

```
 title line  {
   Block
}
```

now let's setup the square function:

```
public static int square(int x) {
         return x*x;
  }
```

new keywords inherit to Java,
otherwise, everything else is the same as in C++

Decisions in c++ (conditional logic)

```
ex)  if(X>0) system.out.println("positive");
     else system.out.println("negative");

     system.out.println();
```

need to account for 0

Aside from printing, the underlying logic remains the same

loops in Java (identitical to c++)

```
    //for loop
ex) int sum=0;
    for(int x=1;x<=10;x++){
       sum= sum+x; // equivalently
                      sum +=x;
    }
```

```
//while loop
 int sum,x=1;
 while(x<=10){
  sum +=x;
 }
```

Arrays, they're a little different between c++ and Java

C++	Java
int data[10];	int data[];
data[2]= 7;	data = new int[10]; // allocate new storage
	data[2]=7;

end of lecture.