Ppt 6장 static, 이것이 자바다p

-static 공유개념을 사용할 때 사용

-static으로 선언된 변수를 접근할때는 new연산자로 객체를 생성할필요없이 클래스명.static변수명으로 접근할수있다(이미 class영역메모리에 올라가있기 때문에)

-static선언으로 안하고 메소드 접근시

**public** **class** Helloworld {

**public** **static** **int** *sum*;

**static** **int** *c*;

**public** **static** **void** main(String[] args) {

String A= "hello";

Helloworld hw = **new** Helloworld(); //객체 생성후 객채.메소드명으로 접근가능

*c* = hw.ps(1,2);

System.***out***.println(A+*c*);

}

**public** **int** ps(**int** a, **int** b) {

*sum* = a+b;

**return** *sum*;

}

}

-singleton

객체하나만을 생성하고 여러명이 하나에 접근하여 메모리사용을 줄일수있다

**public** **class** Helloworld {

**int** a =2;

**private** **static** Helloworld *hw* = **new** Helloworld();

**private** Helloworld() {

}

**static** Helloworld getInstance() {

**return** *hw*;

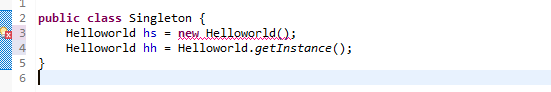
}

**public** **static** **void** main(String[] args) {

//

}

}



다른클래스에서 new연산자로 객채생성불가능하며 getInstance()로 접근하여 값에 접근할수있음을 알수있다

-source ->

-final 로지정시 상수값으로 지정되어 값을 변경할수없다

-다른패키지에서 클래스 접근할 때 import시켜서 접근할수있음( import안할시 해당경로 써서 접근해야함)