Object Oriented Programming JAVA

Dr. Prafulla Kalapatapu

Computer Science Engineering

Mahindra Ecole Centrale

prafulla.kalapatapu@mechyd.ac.in

Exception Handling

Example:

```
class NegSalException extends Exception{
public NegSalException(String s){
System.out.println(s);
} }
class Employee{
int eid,esal;
String ename;
Employee(int i,String en,int s){
eid=i;
ename=en;
esal=s;
}
void display(){
System.out.println(eid+ " "+ename+ " "+esal);
}
}
```

```
class Demo {
public static void  main(String... a) {
Scanner sc=new Scanner(System.in);
int id=sc.nextInt();
String na=sc.nextLine();
int sal=sc.nextInt();
try{
if(sal < 0) {
throw new NegSalException("Sal is in -ve");
}
else {
Employee e =new Employee(id,na,sal);
e.display();
} }
catch(NegSalException e) {
System.out.println("Program end with exception"); } }
}
```

- which exceptions have to handle using try and catch block.

  checked exceptions..

- if not handle checked exceptions.

  It gives compilation error.

- can we compile our program without handling checked exceptions.

  yes, using throws keyword..

- what is the use of throws keyword..

  - programmer explicitly telling to compiler that it is not handling exception using try and catch block.

  - we are giving this responsibility to caller method..

  - if caller method also don't want to handle using try and catch block, it has to throws to JVM finally to avoid compilation problem...

syntax:

void m1() throws Exception