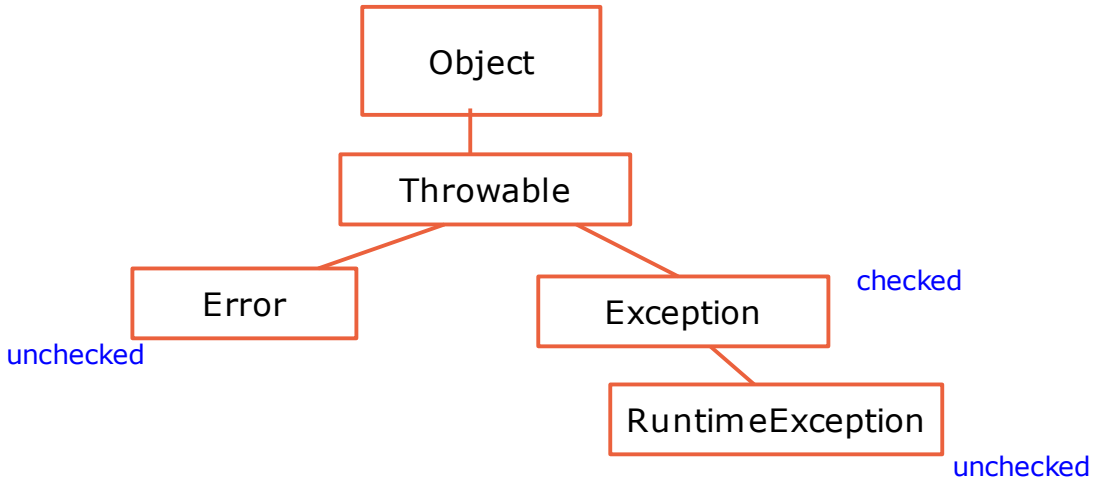
Notes on Exception handling in Java

Java has **Exceptions** and **Errors**. Both are Objects and both extend the class **Throwable**. The object hierarchy is shown below.



Checked exception are those the programmer must list in a "throws" clause. *Unchecked* exceptions are intended for severe, unpredictable errors (such as relating to memory issues) and your code doesn't typically deal with them.

From the programmer's point of view, only checked exceptions are used for error handling. So all programmer-defined exceptions extend the Exception class. Even though RuntimeException extends Exception, it is unchecked.

Here is a generic exception class, called *myException*.

```
/* myException.java */

public class myException extends Exception /*Throwable-ok*/ {
   public String getMessage(){
        return  "Throwing myException ";
   }
   public int negArg(){
        System.out.println("Negative Arg! Returning 100");
        return 100;
   }
   public String f(String e){
        return "myException is throwing back "+e;
   }
}
```

The following class contains methods that throw the myException exception:

```
class exception_example {
   public void f(int i){
        try{
           System.out.println(g(i));
        }catch (Exception e){
           System.out.println(e.getMessage());
        }
   }
   public int g(int i) {
        try{
           if(i<0){
                throw new myException();
           }else
                return i*10;
        }catch (myException e){
           e.negArg();
           return i;
        }
   }
   public static void main(String[] args){
        exception_example t1=new exception_example();
        int input1=Integer.parseInt(args[0]);
        int input2=Integer.parseInt(args[1]);
        t1.f(input1);
        t1.f(input2);
   }
}
```

f() simply calls g()

g(i) returns 10*i if i is >= 0; otherwise it throws an instance of myException – this exception needs to be caught and handled.

The output below shows the result when this program is run with two different inputs.

```
igraine:~/CCN/Programs/Java$ java exception_example 10 9
100
90
igraine:~/CCN/Programs/Java$ java exception_example -10 9
Negative Arg - Please Supply Positive Values only!!!!!!!!
-10
90
igraine:~/CCN/Programs/Java$
```

Note that when the exception is thrown, the program continues execution after printing a message.

We now modify the example to include a method that throws an exception.

```
class exception_example2{
   public void f(int i){
      try{
         System.out.println(g(i));
      }catch (Exception e){
         System.out.println(e.getMessage());
      }
   }
   public int g(int i) {
      try{
         if(i<0){
            throw new myException();
         }else
            return i*10;
      }catch (myException e){
         e.negArg();
         return i;
      }
   }

   public void h() throws myException {
      try{
         System.out.println("--In h()--");
         throw new myException();
      }catch (myException e){
         System.out.println(e.getMessage());
      }
   }

   public static void main(String[] args)//throws myException
   {
      //throws clause or a try-catch block necessary because of h() call.
      exception_example2 t1=new exception_example2();
      int input1=Integer.parseInt(args[0]);
      int input2=Integer.parseInt(args[1]);
      t1.f(input1);
      t1.f(input2);
      try{
         t1.h();
      } catch (myException e){
         e.getMessage();
      }
      t1.f(input2);
   }
}
```

The next page shows output from running the above program.

```
igraine:~/CCN/Programs/Java$ java exception_example2 -2 4
Negative Arg - Please Supply Positive Values only!!!!!!!!
-2
40
--In h()--
Throwing myException
40
igraine:~/CCN/Programs/Java$ java exception_example2 -5 -4
Negative Arg - Please Supply Positive Values only!!!!!!!!
-5
Negative Arg - Please Supply Positive Values only!!!!!!!!
-4
--In h()--
Throwing myException
Negative Arg - Please Supply Positive Values only!!!!!!!!
-4
igraine:~/CCN/Programs/Java$ java exception_example2 4 5
40
50
--In h()--
Throwing myException
50
igraine:~/CCN/Programs/Java$ java exception_example2 4 -9
40
Negative Arg - Please Supply Positive Values only!!!!!!!!
-9
--In h()--
Throwing myException
Negative Arg - Please Supply Positive Values only!!!!!!!!
-9
igraine:~/CCN/Programs/Java$
```