1. **If base class doesn’t throw any exception but child class throws an unchecked exception.**

class Building {

void color()

{

System.out.println("Blue");

}

}

class Room extends Building{

//It throws an unchecked exception

void color() throws NullPointerException

{

System.out.println("White");

}

public static void main(String args[]){

Building obj = new Room();

obj.color();

}

}

1. **If base class doesn’t throw any exception but child class throws an checked exception**

import java.io.\*;

class Building {

void color()

{

System.out.println("Blue");

}

}

class Room extends Building{

void color() throws IOException

{

System.out.println("White");

}

public static void main(String args[]){

Building obj = new Room();

try{

obj.color();

}catch(Exception e){

System.out.println(e);

}

}

}

1. **When base class and child class both throws a checked exception**

import java.io.\*;

class Building {

void color() throws IOException

{

System.out.println("Blue");

}

}

class Room extends Building{

void color() throws IOException

{

System.out.println("White");

}

public static void main(String args[]){

Building obj = new Room();

try{

obj.color();

}catch(Exception e){

System.out.println(e);

}

}

}

1. **When child class method is throwing border checked exception compared to the same method of base class**

package beginnersbook.com;

import java.io.\*;

class Building {

void color() throws IOException

{

System.out.println("Blue");

}

}

class Room extends Building{

void color() throws Exception

{

System.out.println("White");

}

public static void main(String args[]){

Building obj = new Room();

try{

obj.color();

}catch(Exception e){

System.out.println(e);

}

}

}