

Java Core

Excepciones

1. Create a class with a **main()** that throws an object of class **Exception** inside a **try** block. Give the constructor for **Exception** a **String** argument. Catch the exception inside a **catch** clause and print the **String** argument. Add a **finally** clause and print a message to prove you were there.
2. Create your own exception class using the **extends** keyword. Write a constructor for this class that takes a **String** argument and stores it inside the object with a **String** reference. Write a method that prints out the stored **String**. Create a **try-catch** clause to exercise your new exception.
3. Write a class with a method that throws an exception of the type created in Exercise 2. Try compiling it without an exception specification to see what the compiler says. Add the appropriate exception specification. Try out your class and its exception inside a try-catch clause.
4. Define an object reference and initialize it to **null**. Try to call a method through this reference. Now wrap the code in a **try-catch** clause to catch the exception.
5. Create a class with two methods, **f()** and **g()**. In **g()**, throw an exception of a new type that you define. In **f()**, call **g()**, catch its exception and, in the **catch** clause, throw a different exception (of a second type that you define). Test your code in **main()**.
6. Repeat the previous exercise, but inside the **catch** clause, wrap **g()**'s exception in a **RuntimeException**.
7. Create three new types of exceptions. Write a class with a method that throws all three. In **main()**, call the method but only use a single **catch** clause that will catch all three types of exceptions.
8. Write code to generate and catch an **ArrayIndexOutOfBoundsException**.
9. Create your own resumption-like behavior by using a **while** loop that repeats until an exception is no longer thrown.
10. Create a three-level hierarchy of exceptions. Now create a base-class **A** with a method that throws an exception at the base of your hierarchy. Inherit **B** from **A** and override the method so it throws an exception at level two of your hierarchy. Repeat by inheriting class **C** from **B**. In **main()**, create a **C** and upcast it to **A**, then call the method.