

Exception handling:

Unchecked Exception: or Runtime Exception

NULLPOINTER EXCEPTION:

```
public class Main {  
    public static void main(String[] args) {  
        int a=67,b=0;  
        int c=0;  
        try{  
            int arr[]=null;  
            System.out.println(arr[1]);  
            c=a/b;  
        }catch(ArithmeticException e){  
            System.err.println("Exception caught"+e);  
        }catch(NullPointerException e){  
            System.out.println(e);  
        }  
        System.out.println("After catch");  
    }  
}  
  
java.lang.NullPointerException: Cannot load from i  
nt array because "<local4>" is null  
After catch
```

ARITHMETIC EXCEPTION:

```
public class Main {  
    public static void main(String[] args) {  
        int a=67,b=0;  
        int c=0;  
        try{  
            c=a/b;  
        }catch(ArithmeticException e){
```

```

        System.err.println("Exception caught"+e);
    }catch(NullPointerException e){
        System.out.println(e);
    }
    System.out.println("After catch");
}
}

```

```

Exception caughtjava.lang.ArithmeticException: / by zero
After catch

```

NUMBER FORMAT EXCEPTION:

```

public class Main {
    public static void main(String[] args) {
        String invalidNumber = "abc";
        try {
            int number = Integer.parseInt(invalidNumber);
        } catch (NumberFormatException e) {
            System.out.println("NumberFormatException: " + e.getMessage());
        }
        System.out.println("Program continues after handling exception.");
    }
}

```

```

NumberFormatException: For input string: "abc"
Program continues after handling exception.

```

ILLEGAL ARGUMENT EXCEPTION:

```

public class Main {
    public static void main(String[] args) {
        try {
            printAge(-5);
        } catch (IllegalArgumentException e) {

```

```

        System.out.println("Caught an IllegalArgumentException: " +
e.getMessage());
    }
    System.out.println("Program continues after handling exception.");
}
public static void printAge(int age) {
    if (age < 0) {
        throw new IllegalArgumentException("Age cannot be negative");
    }
    System.out.println("Age: " + age);
}
}

```

Run Ask AI 2s on 20:58:04, 05/31 ✓

```

Caught an IllegalArgumentException: Age cannot be
negative
Program continues after handling exception.

```

SECURITY EXCEPTION:

```

public class Main {
    public static void main(String[] args) {
        try {
            performRestrictedAction();
        } catch (SecurityException e) {
            System.out.println("Caught a SecurityException: " + e.getMessage());
        }

        System.out.println("Program continues after handling exception.");
    }

    public static void performRestrictedAction() {
        boolean hasPermission = checkUserPermission();
        if (!hasPermission) {

```

```

        throw new SecurityException("User does not have permission to perform this
action");
    }

    System.out.println("Restricted action performed successfully");
}

public static boolean checkUserPermission() {
    return false;
}
}

```

▼	Run	🔗 Ask AI	2s on 21:12:40, 05/31 ✓
Caught a SecurityException: User does not have per mission to perform this action Program continues after handling exception.			

UNSUPPORTED OPERATION EXCEPTION:

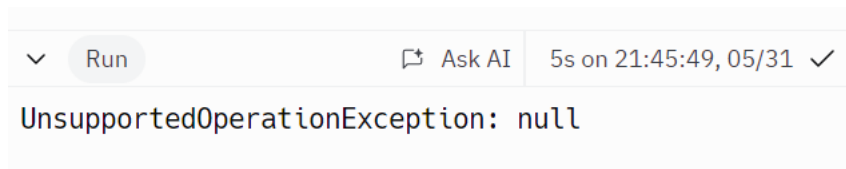
```

import java.util.Arrays;

import java.util.List;

public class Main {
    public static void main(String[] args) {
        String array[] = {"a", "b", "c"};
        List<String> list = Arrays.asList(array);
        try {
            list.add("d");
        } catch (UnsupportedOperationException e) {
            System.err.println("UnsupportedOperationException: " + e.getMessage());
            // Handle the exception here
        }
    }
}

```



Checked Exception: Or Compile Time

MALFORMEDURL EXCEPTION:

```
import java.net.MalformedURLException;
import java.net.URL;

public class Main {
    public static void main(String[] args) {
        String[] urls = {
            "http://valid-url.com",
            "http://invalid-url",
            "https://another-valid-url.com",
            "invalid-url"
        };

        for (String urlString : urls) {
            try {
                URL url = new URL(urlString);
                System.out.println("The URL is valid: " + url);
            } catch (MalformedURLException e) {
                System.out.println("Caught a MalformedURLException for URL: " + urlString);
                System.out.println("Reason: " + e.getMessage());
            }
        }

        System.out.println("Program continues after handling all URLs.");
    }
}
```

```
}  
}
```

```
The URL is valid: http://valid-url.com  
Caught a MalformedURLException for URL: http://invalid-url  
Reason: unknown protocol: http  
The URL is valid: https://another-valid-url.com  
Caught a MalformedURLException for URL: invalid-url  
Reason: no protocol: invalid-url  
Program continues after handling all URLs.
```

PARSE EXCEPTION:

```
public class Main {  
    public static void main(String[] args) {  
        String str = "Hello, world!";  
        try {  
            int num = Integer.parseInt(str);  
            System.out.println("Parsed integer: " + num);  
        } catch (NumberFormatException e) {  
            System.out.println("Parse exception occurred: " + e.getMessage());  
        }  
    }  
}
```

Run Ask AI 3s on 21:33:11, 05/31 ✓

```
Parse exception occurred: For input string: "Hello  
, world!"
```