**일일학습일지** 로고

**제20기 자바&파이썬 웹애플리케이션 전문가 양성과정**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **학습일시** |  | **장소** | kt ds University 211호 | **시 간** | 09:00~18:00 |
| **학습범위** |  | | | | |
| **작 성 자** | 김민성 | | | **강 사** | 장민창 |

|  |  |
| --- | --- |
| **학습안건** |  |

|  |  |
| --- | --- |
| **학습내용** | **내용** |
| 자주 발생하는 예외  NullPointerException - 인스턴스가 널임  ArrayIndexOutOfBoundsException - 배열의 인덱스 벗어남  IndexOutOfBoundsException - List의 인덱스 벗어남  NumberFormatException - 문자를 숫자로 변환하려함  예외처리방법  1. if를 활용  1-1 NullPointerException  인스턴스 참조 전에 인스턴스가 null인지 확인  1-2 IndexOutOfBoundsException  인스턴스 참조 전에 배열의 길이 계산  예외를 읽는 방법(StackTrace)  Exception 발생 위치와 호출 스택이 노출  상위부터 Exception 원인을 파악하면된다.  만든 패키지만 확인(java.lang어쩌구 무시)  예외(Exception) 의 종류  1. (Compile)Exception - try catch 필수  2. RuntimeException - try catch나 if validation체크 둘 중 선택하여 예외처리  try{  예외가 발생할 수도 있는 코드  }catch(Exception e){  e.printStackTrace();  }  finally{  System.out.println("처리완료");  }  1. try 구간 코드는 짧아질수록 좋다.  2. try 구간은 main쓰레드가 처리하지 않고 다른 쓰레드가 처리한다.  Exception 으로 던질 때에는 메소드 뒤에 throws Exception을 붙여줘야 한다.  사용자 예외는 항상 RuntimeException을 상속받아 사용한다. |