

Java

Basic

- 기본 연산, 제어 구조
- 산술 연산, 논리 연산
- 자료형; 기본자료형, 참조형, 인스턴스, 래퍼클래스, 상수
- 비교 연산; 주요 비교 연산자, 참조형 비교, String 비교, 자료형 비교(instanceof 연산자), 논리 연산
- 조건 분기; if, switch, for, while, do-while, break, continue, return
- 객체지향
 - 캡슐화
 - 상속
 - 생성자
 - 다형성
 - 인터페이스
 - 위임
- 제네릭
- 랴다식

Preparing

- 개발환경 구성
 - JDK
 - IDE 혹은 TextEditor
 - Build 도구 (Maven, Gradle) 및 의존성관리
 - 기타 개발 도구

Examples

- String → StringBuilder
- java.util.Date ↔ java.time.Instant
- java.util.Calendar → java.time.Instant
- java.text.SimpleDateFormat
- java.time
 - Local
 - Zoned
 - Offset
- Collections
 - List; ArrayList, LinkedList
 - Set; HashSet, LinkedHashSet
 - Map; HashMap, LinkedHashMap
- 랴다식

(자료형 인수, ...) -> { ... 임의의 처리 ... }

Database

```
import java.sql.*;

...
...
Connection conn = null;
try
{
    Class.forName("DRIVER");
}
catch(ClassNotFoundException e)
{
    e.printStackTrace();
}
try
{
    conn = DriverManager.getConnection("jdbc database", "user", "password");
    Statement stmt = conn.createStatement();
    ResultSet rset = stmt.executeQuery("QUERY");
    while(rset.next())
    {
        // do something
    }
}
catch(SQLException e)
{
    e.printStackTrace();
}
finally
{
    if(conn != null)
    {
        try
        {
            conn.close();
            conn = null;
        }
        catch(SQLException e)
        {
            e.printStackTrace();
        }
    }
}
...
...
```

- `conn.setAutoCommit(true/false);`
- `conn.commit();`
- `conn.rollback();`
- `PreparedStatement stmt = conn.prepareStatement(".... ?");`
 - `stmt.setString(1, "value");`
- DAO Data Access Object
- DTO Data Transfer Object
- JPA Java Persistence API

File I/O

- `FileReader`
- `BufferedReader`
- `Scanner`
- `Files`
- CSV, XML, LOGGER

Tests

- JUnit
 - `@Test`
 - `@Before`
 - `@After`
 - `@BeforeClass`
 - `@AfterClass`
 - `@Ignore`
 - `import static org.junit.Assert.*;`
 - `assertEquals()`
 - `fail()`
 - `assertThat()`
 - `import static org.hamcrest.CoreMatchers.*;`
 - `is()`
- JMockit
- DbUnit

Reserved Words

- data types <list-group>
- `byte`
- `short`
- `int`
- `long`
- `float`
- `double`
- `char`

- boolean
- void

</list-group>

- class & packages <list-group>
- class
- interface
- enum
- package
- import
- extends
- implements
- this
- super
- new

</list-group>

- modifiers <list-group>
- public
- protected
- private
- static
- final
- abstract
- native
- volatile
- transient
- synchronized
- strictfp

</list-group>

- operations <list-group>
- if
- else
- switch
- case
- default
- for
- while
- do
- break
- continue
- return
- instanceof
- assert

</list-group>

- exceptions <list-group>

- try
- catch
- finally
- throw
- throws

</list-group>

- others <list-group>
- const
- goto

</list-group>

References

- 점프 투 자바
- [자바 IDE] IntelliJ vs Eclipse 비교
- MyBatis에서 샷({})과 달러({})의 차이는 무엇일까?
- Java Design Patterns
- Design patterns implemented in Java
- 자바-코드를-조작하는-다양한방법.md

From:

<https://theta5912.net/> - reth

Permanent link:

<https://theta5912.net/doku.php?id=public:computer:java>

Last update: **2022/09/15 23:21**

