**Argument Conversion**

**Writing a code to demonstrate Implicit Conversion**

**package** com.app.junitDemo;

**import** java.time.temporal.ChronoUnit;

**import** org.junit.jupiter.api.Assertions;

**import** org.junit.jupiter.params.ParameterizedTest;

**import** org.junit.jupiter.params.provider.ValueSource;

**public** **class** ArgumentConversionImplicit {

@ParameterizedTest

@ValueSource(strings = "SECONDS") // input data source as String

**public** **void** testwithImplicitArgumentConversion(ChronoUnit chronoUnit) { // method expects a timeUnit not a string

System.***out***.println(chronoUnit.name());

Assertions.*assertNotNull*(chronoUnit.name());

}

}

**Writing a code to demonstrate Explicit Conversion**

**package** com.app.junitDemo;

**import** java.util.concurrent.TimeUnit;

import org.junit.jupiter.api.Assertions;

**import** org.junit.jupiter.params.ParameterizedTest;

import org.junit.jupiter.params.converter.ConvertWith;

import org.junit.jupiter.params.provider.EnumSource;

public class ArgumentConversionexplicitSourceExplicit {

@ParameterizedTest

@EnumSource(TimeUnit.class)

void testCaseWithExplicitArgumentConversion(

@ConvertWith(ToStringArgumentConverter.class) String argument) {

System.***out***.println(TimeUnit.*valueOf*(argument));

Assertions.*assertNotNull*(TimeUnit.*valueOf*(argument));

}

}

**package** com.app.junitDemo;

**import** org.junit.jupiter.api.Assertions;

**import** org.junit.jupiter.params.converter.ArgumentConversionException;

**import** org.junit.jupiter.params.converter.SimpleArgumentConverter;

**public** **class** ToStringArgumentConverter **extends** SimpleArgumentConverter {

@Override

**protected** Object convert(Object source, Class<?> targetType) **throws** ArgumentConversionException {

// **TODO** Auto-generated method stub

Assertions.*assertEquals*(String.**class**, targetType, "Can only convert to String");

**return** String.*valueOf*(source);

}

}