Supplier doesn’t take any input, but returns some common action.

@FunctionalInterface

**public** **interface** Supplier<T> {

T get();

}

==

**import** java.time.LocalDateTime;

**import** java.util.Random;

**import** java.util.function.Supplier;

**public** **class** ExampleOne {

**public** **static** **void** main(String[] args) {

Supplier<String> currentDateTime = () -> {

LocalDateTime currentTime = LocalDateTime.*now*();

**return** currentTime.toString();

};

String test = currentDateTime.get();

System.***out***.println(test);

Supplier<String> randomDay = () -> {

String[] weeks = { "SUNDAY", "MONDAY", "TUESDAY", "WEDNERSDAY", "THURSDAY", "FRIDAY", "SATURDAY" };

**return** weeks[(**int**) (Math.*random*() \* 7)];

};

System.***out***.println(randomDay.get());

System.***out***.println(randomDay.get());

System.***out***.println(randomDay.get());

Supplier<String> otpGenerator = () -> {

String numbers = "0123456789";

Random rndm\_method = **new** Random();

**char**[] otp = **new** **char**[4];

**for** (**int** i = 0; i < 4; i++) {

otp[i] = numbers.charAt(rndm\_method.nextInt(numbers.length()));

}

**return** **new** String(otp);

};

System.***out***.println(otpGenerator.get());

System.***out***.println(otpGenerator.get());

System.***out***.println(otpGenerator.get());

}

}