

## Java 8 DateTime API - 0

Given code of Test.java file:

```
import java.time.*;

public class Test {
    public static void main(String[] args) {
        Instant instant = Instant.now();
        LocalDateTime obj = null; //Line n1
    }
}
```

Which of the following statements will replace null at Line n1 such that Instant object referred by 'instant' is converted to LocalDateTime object?

- A - LocalDateTime.of(instant);
- B - instant.toLocalDateTime();
- C - (LocalDateTime)instant;
- D - instant.atZone(ZoneId.systemDefault()).toLocalDateTime();

# Java 8 DateTime API - 1

Given code of Test.java file:

```
import java.time.*;

public class Test {
    public static void main(String [] args) {
        LocalDateTime t1 = LocalDateTime.now();
        LocalDateTime t2 = LocalDateTime.now();
        System.out.println(Duration.between(t2, t1));
    }
}
```

What will be the result of compiling and executing Test class?

- A - Runtime Exception
- B - Program terminates successfully after displaying the output
- C - Compilation error

## Java 8 DateTime API - 2

Given code of Test.java file:

```
import java.time.LocalDateTime;  
import java.time.format.DateTimeFormatter;  
import java.time.format.FormatStyle;  
  
public class Test {  
    public static void main(String [] args) {  
        LocalDateTime date = LocalDateTime.of(2019, 1, 1, 10, 10);  
        DateTimeFormatter formatter =  
            DateTimeFormatter.ofLocalizedDate(FormatStyle.FULL);  
        System.out.println(formatter.format(date));  
    }  
}
```

Will above code display time part on to the console?

A - No

B - Yes

## Java 8 DateTime API - 3

Given code of Test.java file:

```
import java.time.LocalDate;

public class Test {
    public static void main(String [] args) {
        LocalDate date = LocalDate.ofEpochDay(0);
        System.out.println(date);
    }
}
```

What will be the result of compiling and executing Test class?

- A - 1970-00-01
- B - 1970-01-01
- C - 1970-1-1
- D - Runtime Exception

## Java 8 DateTime API - 4

Given code of Test.java file:

```
import java.time.LocalDateTime;
import java.time.temporal.ChronoUnit;

public class Test {
    public static void main(String [] args) {
        LocalDateTime t1 = LocalDateTime.parse("11:03:15.987");
        System.out.println(t1.plus(22, ChronoUnit.HOURS)
            .equals(t1.plusHours(22)));
    }
}
```

What will be the result of compiling and executing Test class?

A - Runtime Exception

B - false

C - true

## Java 8 DateTime API - 5

Daylight saving time 2018 in United States (US) ends at 4-Nov-2018 2:00 AM. What will be the result of compiling and executing Test class?

```
import java.time.*;

public class Test {
    public static void main(String [] args) {
        LocalDate date = LocalDate.of(2018, 11, 4);
        LocalTime time = LocalTime.of(13, 59, 59);
        ZonedDateTime dt = ZonedDateTime.of(date, time,
            ZoneId.of("America/New_York"));
        dt = dt.plusSeconds(1);
        System.out.println(dt.getHour() + ":" + dt.getMinute() + ":"
            + dt.getSecond());
    }
}
```

A - 13:0:0

B - 14:0:0

C - 12:0:0

## Java 8 DateTime API - 6

Given code of Test.java file:

```
import java.time.*;

public class Test {
    public static void main(String [] args) {
        LocalDate d1 = LocalDate.now();
        LocalDateTime d2 = LocalDateTime.now();
        System.out.println(Duration.between(d1, d2));
    }
}
```

What will be the result of compiling and executing Test class?

- A - Runtime Exception
- B - Program terminates successfully after displaying the output
- C - Compilation error

## Java 8 DateTime API - 7

Daylight saving time 2018 in United States (US) ends at 4-Nov-2018 2:00 AM. What will be the result of compiling and executing Test class?

```
import java.time.*;

public class Test {
    public static void main(String [] args) {
        LocalDate date = LocalDate.of(2018, 11, 4);
        LocalTime time = LocalTime.of(1, 59, 59);
        ZonedDateTime dt = ZonedDateTime.of(date, time,
            ZoneId.of("America/New_York"));
        dt = dt.plusSeconds(1);
        System.out.println(dt.getHour() + ":" + dt.getMinute() + ":"
            + dt.getSecond());
    }
}
```

A - 1:0:0

B - 3:0:0

C - 2:0:0



## Java 8 DateTime API - 8

Given code of Test.java file:

```
import java.time.LocalDate;

public class Test {
    public static void main(String [] args) {
        LocalDate date = LocalDate.ofEpochDay(1);
        System.out.println(date);
    }
}
```

What will be the result of compiling and executing Test class?

- A - 1970-01-01
- B - 1970-1-1
- C - 1970-01-02
- D - 1970-1-2

## Java 8 DateTime API - 9

Given code of Test.java file:

```
import java.time.*;

public class Test {
    public static void main(String [] args) {
        LocalDate date1 = LocalDate.of(2019, 1, 2);
        date1.minus(Period.ofDays(1));
        LocalDate date2 = LocalDate.of(2018, 12, 31);
        date2.plus(Period.ofDays(1));
        System.out.println(date1.equals(date2) + ":" +
            date1.isEqual(date2));
    }
}
```

What will be the result of compiling and executing Test class?

- A - false:false
- B - true:true
- C - false:true
- D - true:false

## Java 8 DateTime API - 10

Given code of Test.java file:

```
import java.time.*;

public class Test {
    public static void main(String [] args) {
        Period period = null;
        System.out.println(period);
    }
}
```

2022 FIFA world cup in Qatar is scheduled to start on 21st Nov 2022. Which of the following statement, if used to replace null, will tell you the period left for 2022 world cup?

- A - Period.between(LocalDate.now(), LocalDateTime.parse("2022-11-21"))
- B - Period.between(LocalDateTime.now(), LocalDateTime.parse("2022-11-21"))
- C - Period.between(LocalDateTime.now(), LocalDate.parse("2022-11-21"))
- D - Period.between(LocalDate.now(), LocalDate.parse("2022-11-21"))

# Java 8 DateTime API - 11

Given code of Test.java file:

```
import java.time.*;
import java.util.Date;

public class Test {
    public static void main(String[] args) {
        Date date = new Date();
        LocalDate localDate = null; //Line n1
    }
}
```

Which of the following two statements will replace null at Line n1 such that Date object referred by 'date' is converted to LocalDate object?

A - (LocalDate)date;

B - date.toLocalDate();

C -

Instant.ofEpochMilli(date.getTime()).atZone(ZoneId.systemDefault()).toLocalDate();

D - date.toInstant().atZone(ZoneId.systemDefault()).toLocalDate();

E - LocalDate.of(date);

## Java 8 DateTime API - 12

Given code of Test.java file:

```
import java.time.*;  
  
public class Test {  
    public static void main(String [] args) {  
        System.out.println(Duration.ofDays(-2));  
    }  
}
```

What will be the result of compiling and executing Test class?

A - PT-48H

B - PT48H

C - P2D

D - P-2D

## Java 8 DateTime API - 13

Given statement:

\_\_\_\_\_ represents date-based amount of time whereas \_\_\_\_\_ represents time-based amount of time.

Which of the following two options correctly fill the blanks in order?

A - Instant, Duration

B - Duration, Instant

C - Period, Duration

D - Duration, Period

## Java 8 DateTime API - 14

Given code of Test.java file:

```
import java.time.Instant;  
  
public class Test {  
    public static void main(String [] args) {  
        System.out.println(Instant.EPOCH);  
    }  
}
```

What will be the result of compiling and executing Test class?

- A - 1970-01-01T00:00:00Z
- B - 1970-01-01T00:00:00.000Z
- C - 1970-01-01T00:00:00
- D - 1970-01-01T00:00:00.000

## Java 8 DateTime API - 15

Given code of Test.java file:

```
import java.time.*;
import java.time.format.DateTimeFormatter;

public class Test {
    public static void main(String [] args) {
        LocalDate date = LocalDate.of(2018, 11, 4);
        DateTimeFormatter formatter =
            DateTimeFormatter.ofPattern("dd-MM-yyyy");

        System.out.println(formatter.format(date).equals(date.format(formatter))
    }
}
```

What will be the result of compiling and executing Test class?

- A - false
- B - Compilation Error
- C - true



## Java 8 DateTime API - 16

Given code of Test.java file:

```
import java.time.LocalDate;

public class Test {
    public static void main(String [] args) {
        LocalDate date = LocalDate.ofEpochDay(-1);
        System.out.println(date);
    }
}
```

What will be the result of compiling and executing Test class?

A - Runtime Exception

B - 1969-12-31

C - 1970-01-01

## Java 8 DateTime API - 17

Given code of Test.java file:

```
import java.time.*;

public class Test {
    public static void main(String [] args) {
        LocalDate date = LocalDate.of(2019, 1, 1);
        LocalDateTime time = LocalDateTime.of(0, 0);
        ZoneId india = ZoneId.of("Asia/Kolkata");
        ZonedDateTime zIndia = ZonedDateTime.of(date, time, india);

        ZoneId us = ZoneId.of("America/Los_Angeles");
        ZonedDateTime zUS = /*INSERT*/;

        System.out.println(Duration.between(zIndia, zUS)); //Line 15
    }
}
```

Current time in India is: 2019-01-01T00:00. Indians have started celebrating New Year. Line 15 prints the duration for which Los Angeles citizens have to wait to celebrate the new year. Which of the following statement replace /\*INSERT\*/ such that Line 15 prints the correct duration?

- A - ZIndia.withZoneSameLocal(us)
- B - Cannot be achieved by just replacing /\*INSERT\*/
- C - ZIndia.withZoneSameInstant(us)

## Java 8 DateTime API - 18

Given code of Test.java file:

```
import java.time.LocalDate;

public class Test {
    public static void main(String [] args) {
        LocalDate date = LocalDate.ofYearDay(2018, 0);
        System.out.println(date);
    }
}
```

What will be the result of compiling and executing Test class?

- A - 2018-1-1
- B - 2017-12-31
- C - 2018-01-01
- D - Runtime Exception

## Java 8 DateTime API - 19

Given code of Test.java file:

```
import java.time.*;

public class Test {
    public static void main(String [] args) {
        LocalDateTime t1 = LocalDateTime.now();
        LocalDateTime t2 = LocalDateTime.now();
        System.out.println(Duration.between(t1, t2));
    }
}
```

What will be the result of compiling and executing Test class?

- A - Program terminates successfully after displaying the output
- B - Runtime Exception
- C - Compilation error

## Java 8 DateTime API - 20

Given code of Test.java file:

```
import java.time.LocalDateTime;

public class Test {
    public static void main(String [] args) {
        LocalDateTime time = LocalDateTime.parse("14:14:59.1111");
        System.out.println(time);
    }
}
```

What will be the result of compiling and executing Test class?

- A - 14:14:59.1111
- B - 14:14:59.111100
- C - 14:14:59.111100000
- D - Runtime Exception

## Java 8 DateTime API - 21

Given code of Test.java file:

```
import java.time.*;

public class Test {
    public static void main(String [] args) {
        Period period = Period.ofWeeks(100);
        System.out.println(period);
    }
}
```

What will be the result of compiling and executing Test class?

- A - p100w
- B - P100W
- C - p700d
- D - P700D

## Java 8 DateTime API - 22

Given code of Test.java file:

```
import java.time.LocalDate;
import java.time.format.DateTimeFormatter;
import java.util.Locale;

public class Test {
    public static void main(String [] args) {
        Locale.setDefault(new Locale("en", "US"));
        LocalDate date = LocalDate.parse("2018-09-10");

        System.out.println(date.format(DateTimeFormatter.ofPattern("dd-
        MMM-yyyy")));
    }
}
```

What will be the result of compiling and executing Test class?

- A - 10-SEPTEMBER-2018
- B - 10-Sep-2018
- C - 10-September-2018
- D - 10-SEP-2018

## Java 8 DateTime API - 23

Given code of Test.java file:

```
import java.time.LocalDateTime;

public class Test {
    public static void main(String [] args) {
        LocalDateTime dt = LocalDateTime.parse("2018-03-16T10:15:30.22");
        System.out.println(dt.toLocalDate() + dt.toLocalTime());
    }
}
```

What will be the result of compiling and executing Test class?

- A - Compilation error
- B - 2018-03-1610:15:30.220
- C - Runtime exception
- D - 2018-03-1610:15:30.22



## Java 8 DateTime API - 24

Below represents ZonedDateTime data:

```
2018-02-01T10:30+05:30[Asia/Kolkata]  
2018-01-31T21:00-08:00[America/Los_Angeles]
```

Do above times represent same instance of time?

A - No

B - Yes

## Java 8 DateTime API - 25

Given code of Test.java file:

```
import java.time.LocalDate;
import java.time.Month;
import java.time.Period;

public class Test {
    public static void main(String [] args) {
        LocalDate startDate = LocalDate.of(2018, Month.MARCH, 1);
        LocalDate endDate = LocalDate.of(2018, Month.MARCH, 11);

        System.out.println(Period.between(endDate, startDate));
    }
}
```

What will be the result of compiling and executing Test class?

- A - P-11D
- B - P10D
- C - P-10D
- D - P11D

## Java 8 DateTime API - 26

Given code of Test.java file:

```
import java.time.LocalDate;

public class Test {
    public static void main(String [] args) {
        LocalDate date = LocalDate.ofYearDay(2018, 32);
        System.out.println(date);
    }
}
```

What will be the result of compiling and executing Test class?

A - Runtime exception

B - 2018-02-02

C - 2018-02-01

D - 2018-2-1

E - 2018-2-2

## Java 8 DateTime API - 27

Given code of Test.java file:

```
import java.time.*;
import java.time.format.DateTimeFormatter;

public class Test {
    public static void main(String [] args) {
        LocalDate date = LocalDate.of(2018, 2, 1);
        DateTimeFormatter formatter = DateTimeFormatter
            .ofPattern("DD'nd day of' uuuu");
        System.out.println(formatter.format(date));
    }
}
```

What will be the result of compiling and executing Test class?

- A - 01nd day of 2018
- B - 32nd day of 2018
- C - 02nd day of 2018
- D - Runtime Exception

## Java 8 DateTime API - 28

Given code of Test.java file:

```
import java.time.LocalDateTime;

public class Test {
    public static void main(String [] args) {
        LocalDateTime dt = LocalDateTime.parse("2018-03-16t10:15:30.22");
        System.out.println(dt.toLocalDate() + " " + dt.toLocalTime());
    }
}
```

What will be the result of compiling and executing Test class?

- A - Compilation error
- B - 2018-03-16 10:15:30.22
- C - Runtime exception
- D - 2018-03-16 10:15:30.220

## Java 8 DateTime API - 29

Given code of Test.java file:

```
import java.time.*;

public class Test {
    public static void main(String [] args) {
        LocalDate date1 = LocalDate.of(2019, 1, 1);
        Duration d = Duration.ofDays(1);
        System.out.println(date1.plus(d));
    }
}
```

What will be the result of compiling and executing Test class?

- A - Runtime exception
- B - Compilation error
- C - 2019-01-02
- D - 2019-01-01

## Java 8 DateTime API - 30

Given code of Test.java file:

```
import java.time.LocalDate;

public class Test {
    public static void main(String [] args) {
        LocalDate ocpCouponPurchaseDate = LocalDate.of(2018, 3, 1);
        System.out.println("Coupon expiry date: "
            + ocpCouponPurchaseDate.plusDays(10));
    }
}
```

What will be the result of compiling and executing Test class?

- A - Compilation error
- B - Coupon expiry date: 2018-3-11
- C - Runtime exception
- D - Coupon expiry date: 2018-03-11

## Java 8 DateTime API - 31

Which of the following classes support time zone?

A - LocalDateTime

B - LocalTime

C - ZonedDateTime

D - LocalDate



## Java 8 DateTime API - 32

Given code of Test.java file:

```
import java.time.LocalDateTime;

public class Test {
    public static void main(String [] args) {
        LocalDateTime dt = LocalDateTime.parse("2018-03-16T10:15:30.22");
        System.out.println(dt);
    }
}
```

What will be the result of compiling and executing Test class?

- A - 2018-03-16T10:15:30.22
- B - 2018-03-16T10:15:30.22
- C - 2018-03-16T10:15:30.220
- D - 2018-03-16T10:15:30.220

## Java 8 DateTime API - 33

Given code of Test.java file:

```
import java.time.*;
import java.time.format.DateTimeFormatter;

public class Test {
    public static void main(String [] args) {
        LocalDate date = LocalDate.of(2018, 11, 4);
        DateTimeFormatter formatter =
            DateTimeFormatter.ofPattern("DD-MM-uuuu");
        System.out.println(formatter.format(date));
    }
}
```

What will be the result of compiling and executing Test class?

A - Runtime Exception

B - Compilation Error

C - 04-11-2018

## Java 8 DateTime API - 34

Given code of Test.java file:

```
import java.time.*;

public class Test {
    public static void main(String [] args) {
        System.out.println(LocalDate.of(2018, 6, 6)
            .plus(Period.parse("P9M")));
    }
}
```

What will be the result of compiling and executing Test class?

- A - 2018-03-06
- B - 2019-03-06
- C - 2018-3-6
- D - 2019-3-6

## Java 8 DateTime API - 35

Given code of Test.java file:

```
import java.time.*;
import java.time.format.DateTimeFormatter;

public class Test {
    public static void main(String [] args) {
        LocalDate valDay = LocalDate.of(2018, 2, 14);
        DateTimeFormatter formatter =
            DateTimeFormatter.ofPattern("DD-MM-uuuu");
        System.out.println(valDay.format(formatter));
    }
}
```

What will be the result of compiling and executing Test class?

- A - 14-02-2018
- B - Runtime Exception
- C - Compilation Error
- D - 45-02-2018