IRC_JAVA_MCQ_DATEANDTIMEAPI

Test Summary

- No. of Sections: 1
- No. of Questions: 15
- Total Duration: 30 min

Section 1 - MCQ

Section Summary

- No. of Questions: 15
- Duration: 30 min

Additional Instructions:

None

Q1. What package is the LocalTime class in?

- java.date
- java.lang
- java.time
- java.util

Q2. How many of the classes Duration, LocalDateTime, and LocalTime have the concept of a time zone?

- One
- Two
- Three
- None

Q3. Which class has a getSeconds() method?

- Only the Duration class
- Only the Period
- Both the Duration and Period classes
- Neither class

Q4. Most of the United States observes daylight savings time on March 12, 2017, by moving the clocks forward an hour at 2 a.m. What does the following code output?

```
1 LocalDate localDate = LocalDate.of( 2017, 3, 12);
2 LocalTime localTime = LocalTime.of( 1, 0);
3 ZoneId zone = ZoneId.of(" America/ New_York");
```

```
4  ZonedDateTime z = ZonedDateTime.of( localDate, localTime, zone);
5  Duration duration = Duration.ofHours( 3);
6  ZonedDateTime later = z.plus( duration);
7  System.out.println( later.getHour());
8
9
```

- 4
- 5
- 6
- None of the above

Q5. What is the output of below code?

```
1  int year = 1874;
2  int month = Month.MARCH;
3  int day = 24;
4  LocalDate date = LocalDate.of( year, month, day);
5  System.out.println( date.isBefore( LocalDate.now()));
6
7
```

- false
- true
- The code does not compile.
- The code compiles but throws an exception at runtime.

Q6. Which correctly fills in the blank to print 2017-01-15?

I. f.format(hatDay)
II. f.formatDate(hatDay)
III. hatDay.format(f)

```
1  import java.util.Date;
2  import java.util.Locale;
3  import java.text.DateFormat;
4  import java.time.*;
5  import java.time.format.DateTimeFormatter;
6  public class Main
7  {
8     public static void main(String[] args)
9     {
10
11        LocalDate date = LocalDate.now();
12        LocalDate hatDay = LocalDate.of(2020, Month.JANUARY, 15);
13        DateTimeFormatter f = DateTimeFormatter.ISO_DATE;
14        System.out.println(_____);
15    }
16 }
17
```

- I
- III

I and III

II and III

Q7. Which of the answer choices is true given the following?

2017-01-07T10: 00-07: 00[America/ Phoenix]
2017-01-07T08: 00-08: 00[America/ Vancouver]

The first date/ time is one hour earlier than the second.

The first date/ time is three hours earlier than the second.

The first date/ time is one hour later than the second.

The first date/ time is three hours later than the second.

Q8. Given that daylight savings time starts on March 12, 2017, at 2 a.m. and clocks jump from 1: 59 a.m. to 03: 00 a.m., which of the following can fill in the blank so the code doesn't throw an exception?

```
1 LocalDate localDate = LocalDate.of( 2017, 3, 12);
2 LocalTime localTime = LocalTime.of(_________);
3 ZoneId zone = ZoneId.of(" America/ New_York");
4 ZonedDateTime z = ZonedDateTime.of( localDate, localTime, zone);
5
6
```

2, 0

3, 0

Either of the above will run without throwing an exception.

Both of these will cause an exception to be thrown.

Q9. What is the result of the following?

```
1  import java.util.Date;
2  import java.util.Locale;
3  import java.text.DateFormat;
4  import java.time.*;
5  import java.time.format.DateTimeFormatter;
6  public class Main
7  {
8     public static void main(String[] args)
9     {
10       LocalDate waffleDay = LocalDate.of( 2017, Month.MARCH, 25);
11       Period period = Period.of( 1, 6, 3);
12       LocalDate later = waffleDay.plus( period);
13       later.plusDays( 1);
14       LocalDate thisOne = LocalDate.of( 2018, Month.SEPTEMBER, 28);
15       LocalDate thatOne = LocalDate.of( 2018, Month.SEPTEMBER, 29);
16       System.out.println( later.isBefore( thisOne) + " " + later.isBefore( thatOne));
17    }
18 }
19
```

false false

false true

true true

The code does not compile.

Q10. Choose the correct option based on this code segment.

```
1 LocalDate babyDOB = LocalDate.of(2015, Month.FEBRUARY, 20);
2 LocalDate now = LocalDate.of(2016, Month.APRIL, 10);
3 System.out.println(Period.between(now, babyDOB).getYears()); // PERIOD_CALC
4
```

The code segment results in a compiler error in the line marked with the comment PERIOD_CALC

The code segment throws a DateTimeException

The code segment prints: 1

The code segment prints: -1

Q11. Which one of the following classes is best suited for storing timestamp values of application events in a file?

java.time.ZoneId class

java.time.ZoneOffset class

java.time.Instant class

java.time.Duration class

Q12. Given the following code segment.
Assume that the time-offset value for the Asia/Singapore time zone from UTC/Greenwich is +08:00. Choose the correct option.

```
1 ZoneId zoneId = ZoneId.of("Asia/Singapore");
2 ZonedDateTime zonedDateTime = ZonedDateTime.of(LocalDateTime.now(), zoneId);
3 System.out.println(zonedDateTime.getOffset());
4
```

this code segment results in throwing DateTimeException

this code segment results in throwing UnsupportedTemporalTypeException

the code segment prints: Asia/Singapore

the code segment prints: +08:00

Q13. Choose the correct option based on this code segment.

```
1  DateTimeFormatter dateFormat = DateTimeFormatter.ISO_DATE;     // DEF
2  LocalDate dateOfBirth = LocalDate.of(2015, Month.FEBRUARY, 31);
3  System.out.println(dateFormat.format(dateOfBirth));           // USE
```

the program gives a compiler error in the line marked with the comment USE

the code segment prints: 2015-02-31

the code segment prints: 2015-02-03

this code segment throws java.time.DateTimeException with the message "Invalid date 'FEBRUARY 31'"

Q14. Consider this code segment:
Which of the following outputs matches the string pattern "EEEE" given in this code segment?

```
1  DateTimeFormatter formatter = DateTimeFormatter.ofPattern("EEEE", Locale.US);
2  System.out.println(formatter.format(LocalDateTime.now()));
```

F

Friday

Sept

September

Q15. How to format date from one form to another?

SimpleDateFormat

DateFormat

SimpleFormat

DateConverter

Answer Key & Solution

Section 1 - MCQ

Q1 java.time

Solution

No Solution

Q2 None

Solution

No Solution

Q3 Only the Duration class

Solution

No Solution

Q4 5

Solution

No Solution

Q5 The code does not compile.

Solution

No Solution

Q6 I and III

Solution

No Solution

Q7 The first date/ time is one hour later than the second.

Solution

No Solution

Q8 Either of the above will run without throwing an exception.

Solution

No Solution

Q9 false true

Solution

No Solution

Q10 The code segment prints: -1

Solution

No Solution

Q11 java.time.Instant class

Solution

No Solution

Q12 the code segment prints: +08:00

Solution

No Solution

Q13 this code segment throws java.time.DateTimeException with the message "Invalid date 'FEBRUARY 31'"

Solution

No Solution

Q14 Friday

Solution

No Solution

Q15 SimpleDateFormat

Solution

No Solution