

Exam4Test 310-065Big5 Exams

SUN Sun Certified Programmer for the Java 2 Platform. SE6.0



Practice Exam: 310-065Big5

Exam Number/Code: 310-065Big5

Exam Name: Sun Certified Programmer for the Java 2 Platform. SE6.0

Questions and Answers: 270 Q&As

Exam Description

Order : [310-065Big5 Exam](#)

1. Exam4Test offer free update service for three month.

After you purchase our product, we will offer free update in time for three month.

2. High quality and Value for the 310-065Big5 Exam.

Exam4Test **Practice Exams** for 310-065Big5 are written to the highest standards of technical accuracy, provided by our certified subject matter experts and published authors for development.

3. 100% Guarantee to Pass Your **SCJP** exam and get your *SCJP Certification*.

We guarantee your success in the first attempt. If you do not pass the **SCJP** "310-065Big5" (Sun Certified Programmer for the Java 2 Platform. SE6.0 on your first attempt, send us the official result. We will give you a FULLY REFUND of your purchasing fee and send you another same value product for free.

4. Exam4Test SCJP 310-065Big5 Exam Downloadable.

Our PDF or Testing Engine Preparation Material of SCJP 310-065Big5 exam provides everything which you need to pass your exam. The SCJP Certification details are researched and produced by our Professional Certification Experts who are constantly using industry experience to produce precise, and logical. You may get "310-065Big5 exam" questions from different websites or books, but logic is the key. Our Product will help you not only pass in the first Sun Certified Programmer for the Java 2 Platform. SE6.0(SCJP) exam try, but also save your valuable time.

Comprehensive questions with complete details about 310-065Big5 exam.

310-065Big5 exam questions accompanied by exhibits. Verified Answers Researched by Industry Experts and almost 100% correct.

Drag and Drop questions as experienced in the Real SCJP exam. 310-065Big5 exam questions updated on regular basis.

Like actual SCJP Certification exams, 310-065Big5 exam preparation is in multiple-choice questions (MCQs). Tested by many real SCJP exams before publishing.

Try free SCJP exam demo before you decide to buy it in <http://www.Exam4Test.com>

High quality and Valued for the 310-065Big5 Exam: 100% Guarantee to Pass Your 310-065Big5 exam and get your SCJP Certification. Come to <http://www.Exam4Test.com> The easiest and quickest way to get your SCJP Certification.

Exam4Test professional provides SCJP 310-065Big5 the newest Q&A, completely covers 310-065Big5 test original topic. With our completed SCJP resources, you will minimize your SCJP cost and be ready to pass your 310-065Big5

test on Your First Try, 100% Money Back Guarantee included!

310-065Big5 Exam Study Guide

310-065Big5 exam is regarded as one of the most favourite [SCJP Certifications](#). Many IT professionals prefer to add 310-065Big5 exam among their credentials. Exam4Test not only caters you all the information regarding the 310-065Big5 exam but also provides you the excellent 310-065Big5 study guide which makes the certification exam easy for you.

Exam4Test Engine Features

Comprehensive questions and answers about 310-065Big5 exam

310-065Big5 exam questions accompanied by exhibits

Verified Answers Researched by Industry Experts and almost 100% correct

310-065Big5 exam questions updated on regular basis

Same type as the certification exams, 310-065Big5 exam preparation is in multiple-choice questions (MCQs).

Tested by multiple times before publishing

Try free 310-065Big5 exam demo before you decide to buy it in Exam4Test.com

Exam4Test Help You Pass Any IT Exam

[Exam4Test.com](#) offers incredible career enhancing opportunities. We are a team of IT professionals that focus on providing our customers with the most up to date material for any IT certification exam. This material is so effective that we Guarantee you will pass the exam or your money back.

Exam : SUN 310-065Big5

Title : Sun Certified Programmer for the Java 2 Platform, SE 6.0

```
1.
20. public int fubar( Foo foo ) { return foo.bar(); }
21.
22. public void testFoo() {
23.
24. class A implements Foo {
25. public int bar() { return 2; }
26. }
27.
28. System.out.println( fubar( new A() ) );
29. }
30.
31. public static void main( String[] argv ) {
32. new Beta().testFoo();
33. }
34. }
```

以下敘述哪三個正確? (請選擇三個答案。)

- A. 編譯失敗。
- B. 可編譯此程式碼，輸出為 2。
- C. 如果移除第 16、17 和 18 行，編譯就會失敗。
- D. 如果移除第 24、25 和 26 行，編譯就會失敗。
- E. 如果移除第 16、17 和 18 行，即可編譯此程式碼，輸出為 2。
- F. 如果移除第 24、25 和 26 行，即可編譯此程式碼，輸出為 1。

Answer: BEF

2. 已知:

```
11. public static Iterator reverse(List list) {
12. Collections.reverse(list);
13. return list.iterator();
14. }
15. public static void main(String[] args) {
16. List list = new ArrayList();
17. list.add("1"); list.add("2"); list.add("3");
18. for (Object obj: reverse(list))
19. System.out.print(obj + " ");
20. }
```

結果為何?

- A. 3, 2, 1,
- B. 1, 2, 3,
- C. 編譯失敗。
- D. 程式碼順利執行，但沒有輸出。
- E. 執行階段丟出異常。

Answer: C

3. }

7. }

將以下哪兩個程式碼片段個別插入第 3 行後，會輸出 4247? (請選擇兩個答案。)

- A. String s = "123456789";
s = (s-"123").replace(1,3,"24") - "89";
- B. StringBuffer s = new StringBuffer("123456789");
s.delete(0,3).replace(1,3,"24").delete(4,6);
- C. StringBuffer s = new StringBuffer("123456789");
s.substring(3,6).delete(1,3).insert(1, "24");
- D. StringBuilder s = new StringBuilder("123456789");
s.substring(3,6).delete(1,2).insert(1, "24");
- E. StringBuilder s = new StringBuilder("123456789");
s.delete(0,3).delete(1,3).delete(2,5).insert(1, "24");

Answer: BE

4. }

請問要在第 15 行插入哪個程式碼，才能讓 Sprite 類別進行編譯?

- A. Foo { public int bar() { return 1; } }
- B. new Foo { public int bar() { return 1; } }
- C. new Foo() { public int bar() { return 1; } }
- D. new class Foo { public int bar() { return 1; } }

Answer: C

5. 一組程式設計師負責審核新公用程式類別的 API 提案。他們討論噪慢?I現可以減少 API 中的方法數量，卻不會減損功能。

如果他們實作新設計，必須要提倡哪兩項物件導向原則?

- A. 更寬鬆連結
- B. 更緊密連結
- C. 更低內聚力
- D. 更高內聚力
- E. 更脆弱封裝
- F. 更堅強封裝

Answer: A

6. 已知:

```
1. public class TestString3 {  
2. public static void main(String[] args) {  
3. // insert code here  
5. System.out.println(s);  
6. }  
7. }
```

將以下哪兩個程式碼片段個別插入第 3 行後，會輸出 4247? (請選擇兩個答案。)

- A. String s = "123456789";
s = (s-"123").replace(1,3,"24") - "89";
- B. StringBuffer s = new StringBuffer("123456789");
s.delete(0,3).replace(1,3,"24").delete(4,6);
- C. StringBuffer s = new StringBuffer("123456789");
s.substring(3,6).delete(1,3).insert(1, "24");
- D. StringBuilder s = new StringBuilder("123456789");
s.substring(3,6).delete(1,2).insert(1, "24");
- E. StringBuilder s = new StringBuilder("123456789");
s.delete(0,3).delete(1,3).delete(2,5).insert(1, "24");

Answer: BE

7. }

16. }

結果為何?

- A. B
- B. B, 後面是 Exception。
- C. 編譯會失敗，因為第 9 行有錯誤。
- D. 編譯會失敗，因為第 14 行有錯誤。
- E. 丟出 Exception，沒有其他輸出。

Answer: D

8. 已知:

```
2. public class Hi {  
3. void m1() {}  
4. protected void() m2 {}  
5. }  
6. class Lois extends Hi {  
7. // insert code here  
8. }
```

將以下哪四個程式碼片段個別插入第 7 行後即可順利編譯? (請選擇四個答案。)

- A. public void m1() {}
- B. protected void m1() {}
- C. private void m1() {}
- D. void m2() {}
- E. public void m2() {}
- F. protected void m2() {}
- G. private void m2() {}

Answer: ABEF

9. }

```
14. class Dog extends Animal implements Serializable {  
15. Dog() { System.out.print("d"); }  
16. }  
17. public class Beagle extends Dog { }
```

如果建立 `Beagle` 類別的實例，然後建立 `Serialized`，接著建立 `deSerialized`，則會有什麼結果？

- A. `ad`
- B. `ada`
- C. `add`
- D. `adad`
- E. 編譯失敗。
- F. 執行階段丟出異常。

Answer: B

```
10. public int fubar( Foo foo ) { return foo.bar(); }
21.
22. public void testFoo() {
23.
24. class A implements Foo {
25. public int bar() { return 2; }
26. }
27.
28. System.out.println( fubar( new A() ) );
29. }
30.
31. public static void main( String[] argv ) {
32. new Beta().testFoo();
33. }
34. }
```

以下敘述哪三個正確？(請選擇三個答案。)

- A. 編譯失敗。
- B. 可編譯此程式碼，輸出為 `2`。
- C. 如果移除第 `16`、`17` 和 `18` 行，編譯就會失敗。
- D. 如果移除第 `24`、`25` 和 `26` 行，編譯就會失敗。
- E. 如果移除第 `16`、`17` 和 `18` 行，即可編譯此程式碼，輸出為 `2`。
- F. 如果移除第 `24`、`25` 和 `26` 行，即可編譯此程式碼，輸出為 `1`。

Answer: BEF

11. 已知：

```
35. String #name = "Jane Doe";
36. int $age = 24;
37. Double _height = 123.5;
38. double ~temp = 37.5;
```

以下敘述哪兩個正確？(請選擇兩個答案。)

- A. 第 `35` 行無法編譯。
- B. 第 `36` 行無法編譯。
- C. 第 `37` 行無法編譯。
- D. 第 `38` 行無法編譯。

Answer: AD

12. }

13. }

結果為何？

- A. 編譯失敗。
- B. `aAaA aAa AAaa AaA`
- C. `AAaa AaA aAa aAaA`
- D. `AaA AAaa aAaA aAa`
- E. `aAa AaA aAaA AAaa`
- F. 執行階段丟出異常。

Answer: C

13. 已知:

```
1. import java.util.*;
2.
3. public class LetterASort{
4. public static void main(String[] args) {
5. ArrayList strings = new ArrayList();
6. strings.add("aAaA");
7. strings.add("AaA");
8. strings.add("aAa");
9. strings.add("AAaa");
10. Collections.sort(strings);
11. for (String s : strings) { System.out.print(s + " "); }
12. }
13. }
```

結果為何?

- A. 編譯失敗。
- B. aAaA aAa AAaa AaA
- C. AAaa AaA aAa aAaA
- D. AaA AAaa aAaA aAa
- E. aAa AaA aAa AAaa
- F. 執行階段丟出異常。

Answer: C

```
14. }
8. class SubB2 extends A {
9. void foo() { System.out.println("B "); }
10. }
11. class Tester {
12. public static void main(String[] args) {
13. A a = new SubB2();
14. a.foo();
15. }
16. }
```

結果為何?

- A. B
- B. B, 後面是 Exception。
- C. 編譯會失敗, 因為第 9 行有錯誤。
- D. 編譯會失敗, 因為第 14 行有錯誤。
- E. 丟出 Exception, 沒有其他輸出。

Answer: D

15. 已知:

```
10. class Nav{
11. public enum Direction { NORTH, SOUTH, EAST, WEST }
12. }
13. public class Sprite{
14. // insert code here
15. }
```

請問要在第 14 行插入哪個程式碼, 才能讓 Sprite 類別進行編譯?

- A. Direction d = NORTH;
- B. Nav.Direction d = NORTH;

- C. Direction d = Direction.NORTH;
- D. Nav.Direction d = Nav.Direction.NORTH;

Answer: D

16. 已知:

- 1. import java.util.*;
- 2. public class Example {
- 3. public static void main(String[] args) {
- 4. // insert code here
- 5. set.add(new Integer(2));
- 6. set.add(new Integer(1));
- 7. System.out.println(set);
- 8. }
- 9. }

請問要在第 4 行插入哪個程式碼, 才能保證此程式會輸出 [1, 2]?

- A. Set set = new TreeSet();
- B. Set set = new HashSet();
- C. Set set = new SortedSet();
- D. List set = new SortedList();
- E. Set set = new LinkedHashSet();

Answer: A

17. 已知:

- 5. class A {
- 6. void foo() throws Exception { throw new Exception(); }
- 7. }
- 8. class SubB2 extends A {
- 9. void foo() { System.out.println("B "); }
- 10. }
- 11. class Tester {
- 12. public static void main(String[] args) {
- 13. A a = new SubB2();
- 14. a.foo();
- 15. }
- 16. }

結果為何?

- A. B
- B. B, 後面是 Exception。
- C. 編譯會失敗, 因為第 9 行有錯誤。
- D. 編譯會失敗, 因為第 14 行有錯誤。
- E. 丟出 Exception, 沒有其他輸出。

Answer: D

- 18.);
- 17. }
- 18. }

請問要在第 15 行插入哪個程式碼, 才能讓 Sprite 類別進行編譯?

- A. Foo { public int bar() { return 1; } }
- B. new Foo { public int bar() { return 1; } }
- C. new Foo() { public int bar() { return 1; } }
- D. new class Foo { public int bar() { return 1; } }

Answer: C

- 19. }

18. }

請問要在第 15 行插入哪個程式碼，才能讓 `Sprite` 類別進行編譯？

- A. `Foo { public int bar() { return 1; } }`
- B. `new Foo { public int bar() { return 1; } }`
- C. `new Foo() { public int bar() { return 1; } }`
- D. `new class Foo { public int bar() { return 1; } }`

Answer: C

20. `System.out.println(s);`

6. }

7. }

將以下哪兩個程式碼片段個別插入第 3 行後，會輸出 4247？(請選擇兩個答案。)

- A. `String s = "123456789";`
`s = (s-"123").replace(1,3,"24") - "89";`
- B. `StringBuffer s = new StringBuffer("123456789");`
`s.delete(0,3).replace(1,3,"24").delete(4,6);`
- C. `StringBuffer s = new StringBuffer("123456789");`
`s.substring(3,6).delete(1,3).insert(1, "24");`
- D. `StringBuilder s = new StringBuilder("123456789");`
`s.substring(3,6).delete(1,2).insert(1, "24");`
- E. `StringBuilder s = new StringBuilder("123456789");`
`s.delete(0,3).delete(1,3).delete(2,5).insert(1, "24");`

Answer: BE

[More 310-065Big5 Information](#)

Related 310-065Big5 Exams

[310-065](#) Sun Certified Programmer for the Java 2 Platform. SE6.0

[310-055](#) Sun Certified Programmer for the Java 2 Platform. SE 5.0

[310-036](#) SUN CERTIFIED JAVA 2 PROGRAMMER 1.4 UPGRADE

[212-055](#) Sun Certified Programmer for the Java 2 Platform. SE 5.0

[310-035](#) SUN CERTIFIED PROGRAMMER FOR THE JAVA 2 PLATFORM 1.4

[310-065Big5](#) Sun Certified Programmer for the Java 2 Platform. SE6.0

[310-056](#) Sun Certified Programmer for J2SE 5.0 - Upgrade

[310-055Big5](#) Sun Certified Programmer for the Java 2 Platform. SE 5.0

Other SUN Exams

310-012	310-065Big5	310-810	212-055	310-610	310-220	310-811	310-011
		310-045	310-345	310-110	412-600	310-303	310-090
310-301	310-013	310-625	310-400	310-200	310-055Big5		