Тест

Для выполнения теста по теме Thread.JMM.SyncVolatile удовлетворительно пройдите Тест 2 нажмите "Start Quiz"

Start Quiz

Тест

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 1 of 16

1. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private int value = 0;
    public synchronized int getValue() {return value;}
    public synchronized void setValue(int v) {this.value = v;}
}
```

- ~~не нужен~~
- ~~volatile~~
- ~~synchronized~~
- ~~volatile~~
- synchronized

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 2 of 16

2. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private volatile int value = 0;
    public synchronized int getValue() {return value;}
    public void inc() {value++;}
}
```

- ~~volatile~~
- ~~volatile~~
- ~~synchronized~~
- ~~volatile~~
- ~~volatile~~

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 3 of 16

3. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private volatile int value = 0;
    public synchronized int getValue() {return value;}
    public void inc() {value++;}
}
```

- ~~volatile~~
- ~~volatile~~
- ~~synchronized~~
- ~~volatile~~
- ~~volatile~~

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 4 of 16

4. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private volatile int value = 0;
    public synchronized int getValue() {return value;}
    public void setValue(int v) {this.value = v;}
}
```

- ~~не нужен~~
- ~~volatile~~
- ~~synchronized~~
- ~~volatile~~
- ~~volatile~~

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 5 of 16

5. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private volatile int value = 0;
    public int getValue() {return value;}
    public void setValue(int v) {this.value = v;}
}
```

- ~~volatile~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 6 of 16

6. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private int value = 0;
    public int getValue() {return value;}
    public void setValue(int v) {this.value = v;}
}
```

- ~~volatile~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 7 of 16

7. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private int value = 0;
    public synchronized int getValue() {return value;}
    public synchronized void inc() {value++;}
}
```

- ~~volatile~~
- ~~не нужен~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 8 of 16

8. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private int value = 0;
    public synchronized int getValue() {return value;}
    public synchronized void inc() {value++;}
}
```

- ~~volatile~~
- ~~volatile~~
- ~~не нужен~~
- ~~volatile~~
- ~~volatile~~

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 9 of 16

9. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private int value = 0;
    public synchronized int getValue() {return value;}
    public synchronized void inc() {value++;}
}
```

- ~~volatile~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 10 of 16

10. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private int value = 0;
    public int getValue() {return value;}
    public synchronized void setValue(int v) {this.value = v;}
}
```

- ~~volatile~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 11 of 16

11. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private int value = 0;
    public synchronized int getValue() {return value;}
    public synchronized void setValue(int v) {this.value = v;}
}
```

- ~~volatile~~
- ~~не нужен~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 12 of 16

12. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private int value = 0;
    public synchronized int getValue() {return value;}
    public void setValue(int v) {this.value = v;}
}
```

- ~~volatile~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 13 of 16

13. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private volatile int value = 0;
    public int getValue() {return value;}
    public void inc() {value++;}
}
```

- ~~volatile~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 14 of 16

14. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private int value = 0;
    public synchronized int getValue() {return value;}
    public void setValue(int v) {this.value = v;}
}
```

- ~~volatile~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 15 of 16

15. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private int value = 0;
    public int getValue() {return value;}
    public synchronized void inc() {value++;}
}
```

- ~~volatile~~
- ~~не нужен~~
- ~~volatile~~
- ~~volatile~~
- ~~volatile~~

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Answered Review

Review question

Quiz summary

Question 16 of 16

16. Question

Как обеспечить доступ к данным класса с потокобезопасностью(thread-safe) (возможностью безопасно использовать один экземпляр класса одновременно в нескольких потоках)?

```
public class IncDecDec {
    private int value = 0;
    public synchronized int getValue() {return value;}
    public void setValue(int v) {this.value = v;}
}
```