

000 32-000 00 64-000 000 00 0000 0000000000 0000

“000000 0000 64-000 000 0000000000 00 32-000 000”समस्या को हल करने के लिए, आपको अपने 0000 0000000 0000000 (000) और 0000000 इंस्टॉलेशन के आर्किटेक्चर को मिलाने की जरूरत है। यह समस्या तब होती है जब आपका 0000000 64-बिट 000 लाइब्रेरी का उपयोग करने की कोशिश करता है, लेकिन आपका 000 32-बिट है, जिससे वे असंगत हो जाते हैं। नीचे दिए गए कदमों से आप अपने 0000 आर्किटेक्चर को जांच सकते हैं और समस्या को सुलझा सकते हैं।

कदम 1: जांचें कि आपके पास 32-बिट या 64-बिट 0000 है

आप इनमें से किसी एक तरीके से जांच सकते हैं कि आपका 000 32-बिट या 64-बिट है:

तरीका 1: java -version का उपयोग करके

1. कमांड प्रॉम्प्ट खोलें:

- 00000000: Win + R दबाएं, cmd टाइप करें और 000000 दबाएं।
- 000000/0000: टर्मिनल खोलें।

2. नीचे दिए गए कमांड टाइप करें और 000000 दबाएं:

```
java -version
```

3. आउटपुट जांचें:

- **64-बिट 0000**: आउटपुट में “64-0000” शामिल होगा, उदाहरण के लिए:

```
java version "1.8.0_311"
Java(TM) SE Runtime Environment (build 1.8.0_311-b11)
Java HotSpot(TM) 64-Bit Server VM (build 25.311-b11, mixed mode)
```

- **32-बिट 0000**: आउटपुट में “64-0000” नहीं होगा, उदाहरण के लिए:

```
java version "1.8.0_311"
Java(TM) SE Runtime Environment (build 1.8.0_311-b11)
Java HotSpot(TM) Client VM (build 25.311-b11, mixed mode)
```

तरीका 2: java -d64 -version का उपयोग करके

1. कमांड प्रॉम्प्ट में, नीचे दिए गए कमांड टाइप करें:

```
java -d64 -version
```

2. परिणाम का अर्थ निकालें:

□ **64-बिट** □□□: यह ऊपर दिए गए तरह के वर्जन जानकारी दिखाएगा।

□ **32-बिट** □□□: आप एक त्रुटि देखेंगे, जैसे:

```
Unrecognized option: -d64
```

```
Error: Could not create the Java Virtual Machine.
```

```
Error: A fatal exception has occurred. Program will exit.
```

अगर आपका □□□ 32-बिट है (जैसे कि त्रुटि सुझाती है), तो कदम 2 पर आगे बढ़ें ताकि असंगति को ठीक किया जा सके।

कदम 2: आर्किटेक्चर असंगति को सुलझाएं

इस समस्या को सुलझाने के लिए आपको दो विकल्प हैं:

विकल्प 1: 64-बिट □□□ इंस्टॉल करें (सिफारिश की जाती है)

□ **क्यों?** 64-बिट □□□ प्रदर्शन के लिए बेहतर है और बड़े मेमोरी आवंटन का समर्थन करता है, खासकर अगर आपका सिस्टम 64-बिट है।

□ **कदम:**

1. एक 64-बिट □□□ या □□□ को एक विश्वसनीय स्रोत से डाउनलोड करें, जैसे:

- □□□□□□ □□□

- □□□□□□□□

2. इसे इंस्टॉल करें और इंस्टॉलेशन पथ नोट करें (उदाहरण के लिए, C:\Program Files\Java\jdk1.8.0_XXX\bin\javaw.exe).

3. □□□□□□□□ को 64-बिट □□□ का उपयोग करने के लिए कॉन्फिगर करें:

- अपने □□□□□□□□ इंस्टॉलेशन फोल्डर में eclipse.ini फ़ाइल खोलें (उदाहरण के लिए, C:\eclipse\eclipse.ini).

- -vmargs सेक्शन से पहले इन लाइनों को जोड़ें:

```
-vm
```

```
C:/Program Files/Java/jdk1.8.0_XXX/bin/javaw.exe
```

- आगे की स्लैश (/) का उपयोग करें और सुनिश्चित करें कि पथ आपके इंस्टॉलेशन से मेल खाता है।

4. फ़ाइल को सेभ करें और □□□□□□□□ को पुनः शुरू करें।

विकल्प 2: 32-बिट □□□□□□□□ पर स्विच करें

□ **क्यों?** अगर आप 32-बिट □□□ को रखना चाहते हैं, तो आप 32-बिट □□□□□□□□ के साथ संगत □□□ लाइब्रेरी का उपयोग कर सकते हैं।

□ **कदम:**

1. □□□□□□□□ डाउनलोड पेज से 32-बिट संस्करण का डाउनलोड करें।

2. इसे इंस्टॉल करें और अपने मौजूदा 32-बिट □□□ के साथ इसे लॉन्च करें।

याद रखने के लिए मुख्य बिंदु

- **आर्किटेक्चर को मिलाएं:** आपके `OS` और `JRE` को एक ही आर्किटेक्चर होना चाहिए:
 - 64-बिट `OS` को 64-बिट `JRE` की जरूरत है।
 - 32-बिट `OS` को 32-बिट `JRE` की जरूरत है।
- **पथ दोबारा जांचें:** अगर आप `eclipse.ini` को संपादित करते हैं, तो सुनिश्चित करें कि `javaw.exe` के लिए पथ सही है।
- **सिस्टम संगतता:** अगर आपका ऑपरेटिंग सिस्टम 64-बिट है, तो 64-बिट `OS` पर अपग्रेड करना लंबे समय के लिए बेहतर समाधान है।

इन कदमों का पालन करके, आप त्रुटि को सुलझा देंगे और `JRE` को सुलभ रूप से चलाने में सफल होंगे!