********** OUTPUT **********

```
Welcome to Online IDE!! Happy Coding :)
Exception in thread "main"
java.lang.NullPointerException
    at analyseString.<init>(Main.java:24)
    at Main.main(Main.java:15)


** Process exited - Return Code: 1 **
```

The lines affected are highlighted below:

*** CODE ***

```
/*
Online Java - IDE, Code Editor, Compiler

Online Java is a quick and easy tool that helps you to build, compile, test your programs
online.
*/

public class Main
{
   public static void main(String[] args) {
      System.out.println("Welcome to Online IDE!! Happy Coding :)");

      String sample = "Hello world here";

      analyseString as = new analyseString (sample);

   }
}


class analyseString
{
   private String sample;
   int end = sample.length()-1;  // this is used as excluded delimiterfor substring in order to
extract the word.

   String storeWords = null;

   public analyseString(String sample)
   {
      this.sample=sample;
      findBlankspaces();
   }

   public void findBlankspaces()
```

```
   {
      for (int i=sample.length()-1; i<=0;i--)
      {
         if (sample.charAt(i)==' ')   //if it finds a blank space
         {

            //this word will now have to be stored in String

            if (storeWords==null)  // i.e this is the first occurence of blank space
            {
               if (i!=sample.length()-1)  // to ensure no over run (only instance would be if the
string is total blank)
               {
               storeWords=""; // the string will have to be erased to ensure null is not printed in
concatenation

               storeWords = sample.substring(i+1,end) + Character.toString(end);  // this will
extract the word from the sample....
            // issue is it will exclude end character... so need to a way to manually append extra
character
            // hence Character.toString(end)

            System.out.println("Word extracted: " + storeWords);

            end = i;   // end will now be where it found blank space.. and it will exclude this on
substring going forward which is requirement
               }
            }

            else
            {
               storeWords = storeWords + sample.substring(i+1,end);

               System.out.println("Word extracted: " + sample.substring(i+1,end));

            }

           }
      }

      // this will now print out the string reversed

      System.out.println("These are the words reversed:" + storeWords);

   }

}
```