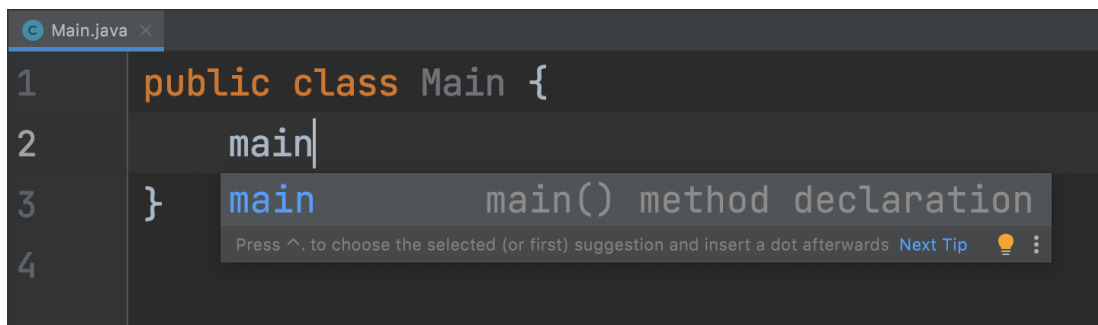




Задание 1. Создать новый проект DateTime и переписать в него код

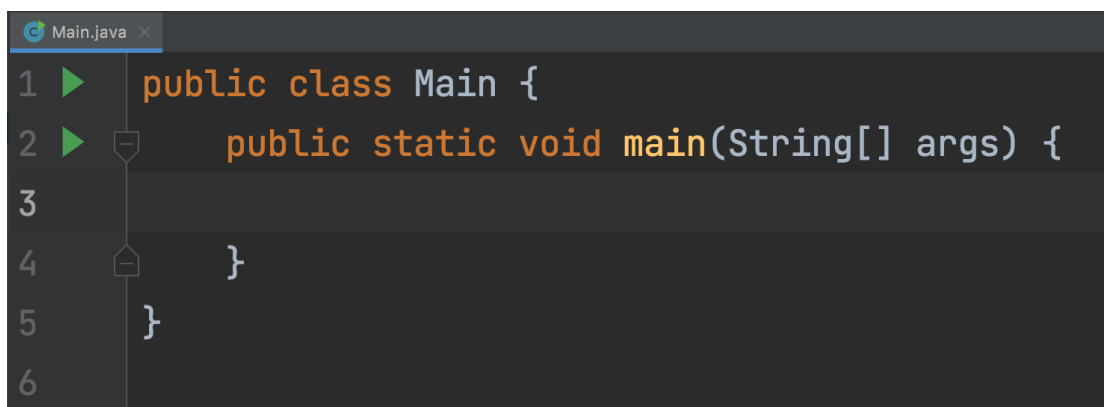
1. Создайте в среде разработки новый проект с именем DateTime.
2. Создайте в этом проекте в папке src класс Main.
3. В коде внутри фигурных скобок напишите слово main и подождите пару секунд, пока не появится предложение автоматической генерации кода:



```
1 public class Main {  
2     main|  
3 } main main() method declaration  
4
```

Press ^, to choose the selected (or first) suggestion and insert a dot afterwards [Next Tip](#)  

4. Кликните по появившемуся предложению дважды. В результате в классе Main будет автоматически создан метод main. Код будет выглядеть следующим образом:

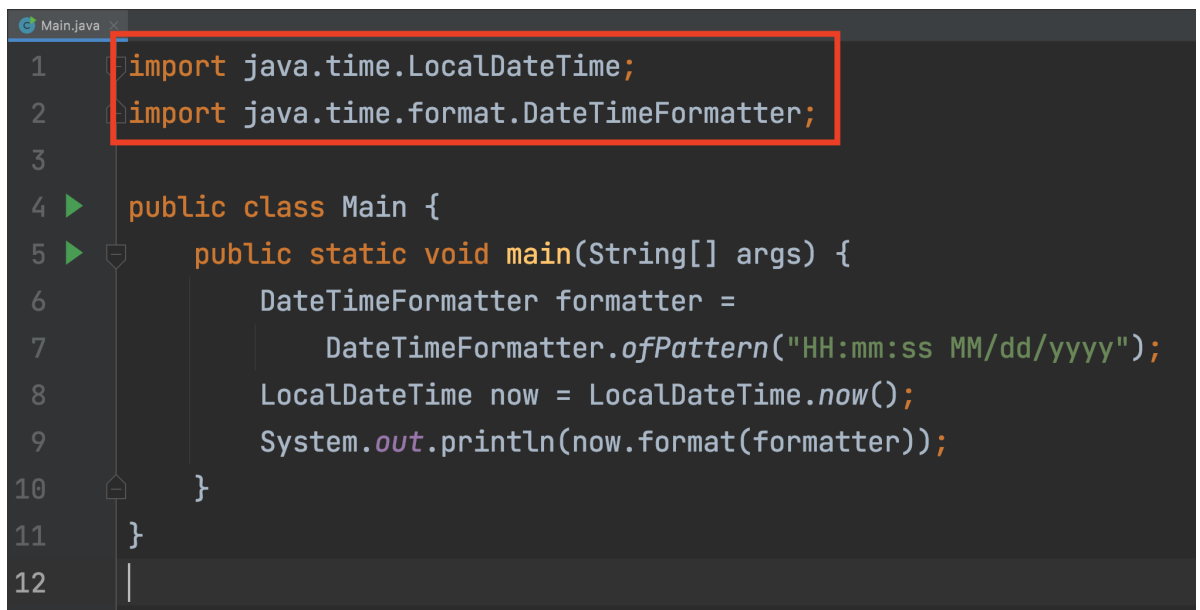


```
1 public class Main {  
2     public static void main(String[] args) {  
3  
4     }  
5 }  
6
```

5. Внутри метода main (между открывающей и закрывающей фигурными скобками метода, в примере выше — в строку 3) перепишите следующий код:

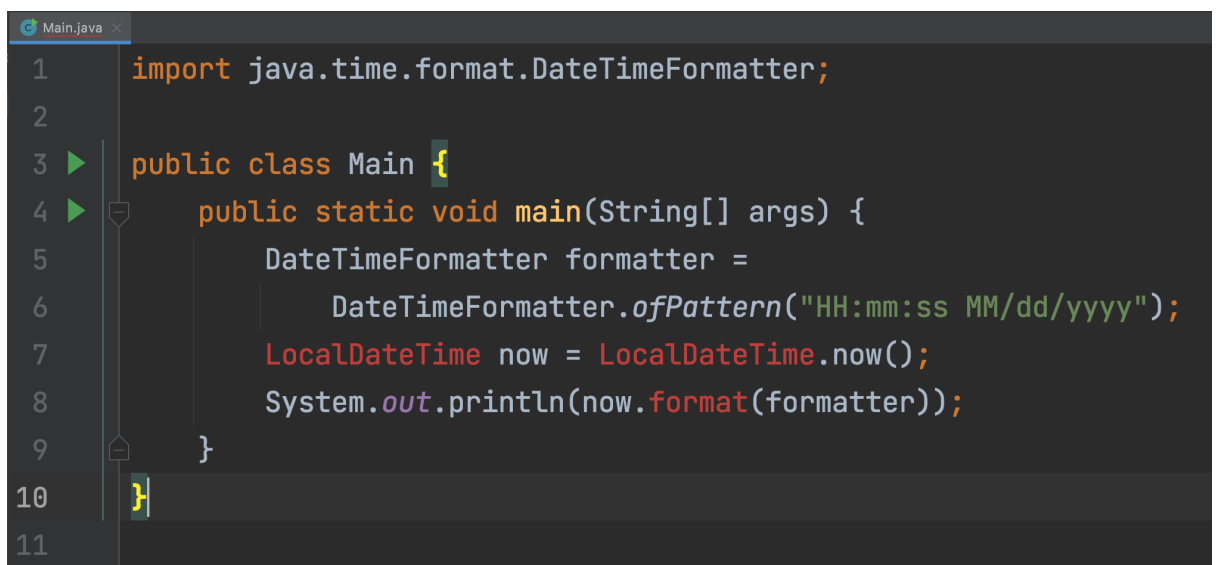
```
DateTimeFormatter formatter =  
    DateTimeFormatter.ofPattern("HH:mm:ss MM/dd/yyyy");  
LocalDateTime now = LocalDateTime.now();  
System.out.println(now.format(formatter));
```

6. В процессе переписывания кода над именем класса должны будут появиться две строки кода:



```
1 import java.time.LocalDateTime;
2 import java.time.format.DateTimeFormatter;
3
4 public class Main {
5     public static void main(String[] args) {
6         DateTimeFormatter formatter =
7             DateTimeFormatter.ofPattern("HH:mm:ss MM/dd/yyyy");
8         LocalDateTime now = LocalDateTime.now();
9         System.out.println(now.format(formatter));
10    }
11 }
12
```

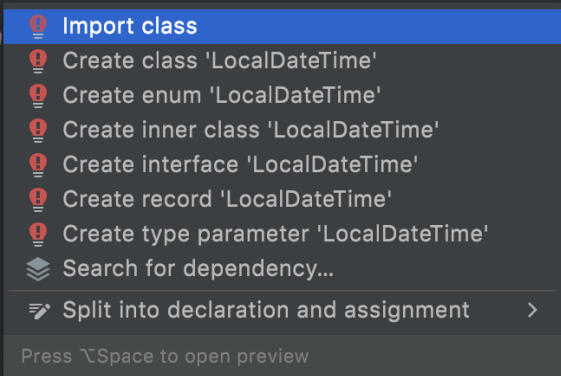
Эти строки могли не появиться (или одна из них не появилась), и какие-то фрагменты кода выделились красным цветом, например, так:



```
1 import java.time.format.DateTimeFormatter;
2
3 public class Main {
4     public static void main(String[] args) {
5         DateTimeFormatter formatter =
6             DateTimeFormatter.ofPattern("HH:mm:ss MM/dd/yyyy");
7         LocalDateTime now = LocalDateTime.now();
8         System.out.println(now.format(formatter));
9     }
10 }
11
```

В этом случае необходимо кликнуть по первому слову, выделенному красным (в примере выше — по `LocalDateTime` в строке 7), нажать клавиши `<Ctrl+Alt>` (в Mac OS — `⌘+⌥`), после чего либо ошибка будет автоматически исправлена, либо выпадет такое меню:

```
3  ▶ public class Main {
4  ▶  ▶ public static void main(String[] args)
5      DateTimeFormatter formatter =
6          DateTimeFormatter.ofPattern("H
7      ⚠ LocalDateTime now = LocalDateTime.
8      System.o
9  ▶ }
10 }
11
```



The context menu is open over the red text 'LocalDateTime' on line 7. The menu options are: 'Import class' (highlighted in blue), 'Create class 'LocalDateTime'', 'Create enum 'LocalDateTime'', 'Create inner class 'LocalDateTime'', 'Create interface 'LocalDateTime'', 'Create record 'LocalDateTime'', 'Create type parameter 'LocalDateTime'', 'Search for dependency...', 'Split into declaration and assignment' (with a right arrow), and 'Press `\Space` to open preview'.

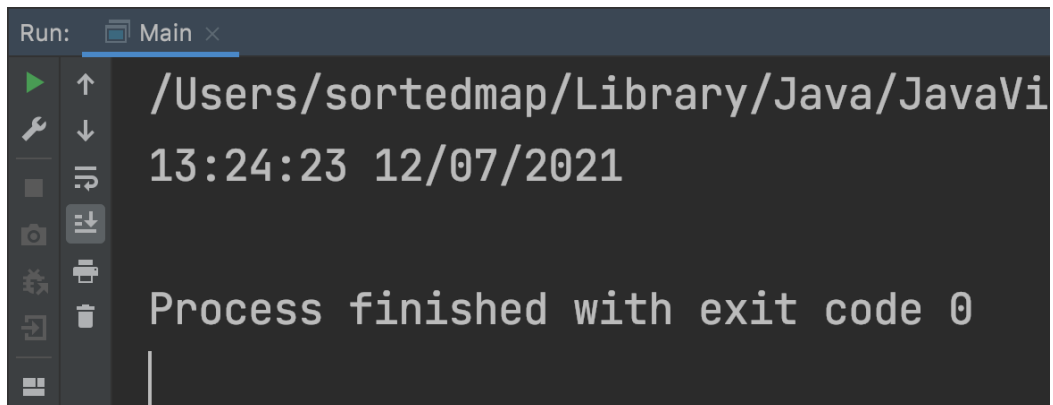
В этом меню необходимо нажать на пункт `Import class`, и ошибка будет исправлена.

7. Если какие-то фрагменты кода всё ещё будут выделены красным, повторите с ними те же действия, которые показаны выше.
8. Если ошибку исправить не удалось, возможно, вы допустили опечатку в коде. Приведите свой код в соответствие с кодом на изображении:

```
Main.java
1  import java.time.LocalDateTime;
2  import java.time.format.DateTimeFormatter;
3
4  ▶ public class Main {
5  ▶  ▶ public static void main(String[] args) {
6      DateTimeFormatter formatter =
7          DateTimeFormatter.ofPattern("HH:mm:ss MM/dd/yyyy");
8      LocalDateTime now = LocalDateTime.now();
9      System.out.println(now.format(formatter));
10 }
11 }
12
```

После этого все сообщения об ошибках и выделения красным исчезнут.

9. Запустите код и убедитесь в том, что в консоль выводится текущая дата.
Пример такого вывода:



The screenshot shows the 'Run' console of an IDE. The title bar indicates the active window is 'Main'. The console output consists of three lines: the first line is the file path `/Users/sortedmap/Library/Java/JavaVi`, the second line is the timestamp `13:24:23 12/07/2021`, and the third line is the message `Process finished with exit code 0`. A vertical cursor is visible at the end of the third line.

```
Run: Main ×  
/Users/sortedmap/Library/Java/JavaVi  
13:24:23 12/07/2021  
Process finished with exit code 0
```