Tips and Tricks

What we'll cover:

- Tips and tricks on code selections
- How to move blocks of code around
- Working in columnar format
- Code formatting
- Views

Before we end the book, I'd like to leave you with a couple more tips and some tricks that, I think, are really helpful.

Selecting Code

Are you still using the mouse to highlight code blocks? That's alright. But IntelliJ gives us another way to (more quickly) work with blocks of code.

You can use the **Option + Up arrow** (macOS) or **CTRL + W** (Windows, Linux) to extend your text selection and include ever-increasing, valid blocks of code. Use the **Option + Down arrow** (macOS) or **CTRL + Shift + W** (Windows, Linux) to decrease the selection. This key is actually smart enough to select the proper demarcations for the code blocks — Figure B-1 shows a snapshot when I was trying out this feature.

Figure B-1. *Selecting blocks of code*

Columnar format

If, for some reason, you need to work with text in a columnar format — like shown in Figure B-2 — IntelliJ has got you covered, too.
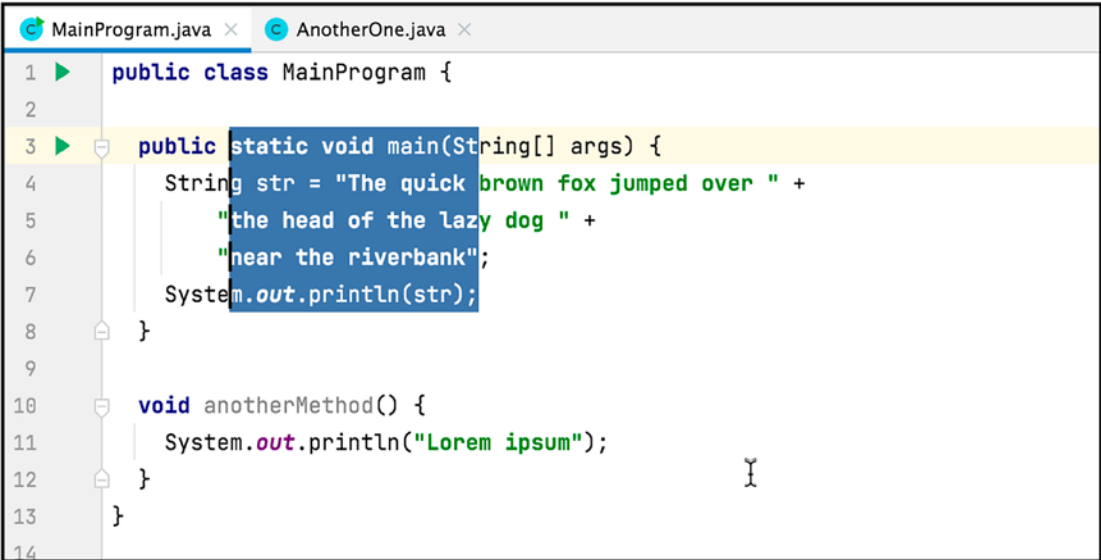

Figure B-2. *Column select*

You need to toggle IntelliJ's Column Select mode; by default, it's off. You've got to turn it on. There are a couple of ways to do it. You can use the keys **Command + Shift + A** (macOS) or **CTRL + Shift + A** (Windows, Linux) — you'll remember these keys as the Find Action keys. Or, just use the ever-reliable and unforgettable Double Shift key shortcut (Search everywhere); then, type "column select" (as shown in Figure B-3).
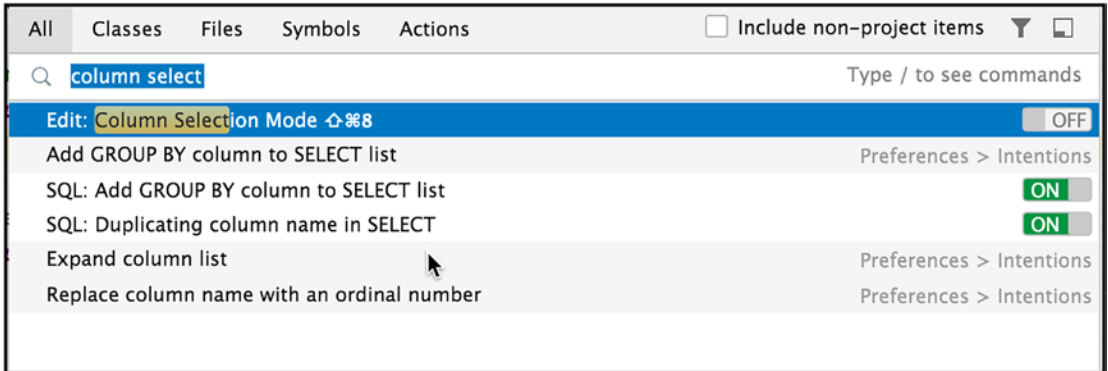

Figure B-3. *Column Select mode*

Moving code blocks

If you've been copying and pasting blocks of code or statements just to move them around, here's another handy trick in IntelliJ.

You can use the keys **Shift + Command + Up Arrow** (macOS) or **CTRL + Shift + Up Arrow** (Linux, Windows) to move whole statements (even multi-line ones) or entire methods.

String Manipulation

Did you ever have to work with a long string in your code? That was a rhetorical question. Of course, you have. Remember the fun of checking if each line of string was properly enclosed in double-quotes? And checking to see if you actually concatenated them properly — like this.

```
public static void main(String[] args) {
  String str = "The quick brown fox ";
     str += "jumped over the head ";
     str+= "of the lazy dog";
  System.out.println(str);
}
```

Well, with IntelliJ, you don't have to manually concatenate long strings over many lines. You can start with a long string, like in the code snippet below.

```
public static void main(String[] args) {
  String str = "The quick brown fox jumped over the head of the lazy dog
  near the riverbank";
  System.out.println(str);
}
```

Then, just place the caret symbol wherever you like within the string, then press ENTER. IntelliJ IDEA will automatically close off the String on the line above, add the '+' symbol for concatenation, and reopen the String on the line below. It will result in something like this.

```
public static void main(String[] args) {
  String str = "The quick brown " +
     "fox jumped over the head of " +
     "the lazy dog near " +
     "the riverbank";
  System.out.println(str);
}
```

On the other hand, if you'd like to rejoin the lines together, try the keys **CTRL + Shift + J** (same keys on macOS, Windows, and Linux).

These keyboard shortcuts are useful when working with block comments and multiline strings.

Formatting Code

If you've been manually deleting and adding tabs to prettify your code, IntelliJ has a neat little trick for quickly formatting your code. Consider Figure B-4; it's obviously poorly formatted.
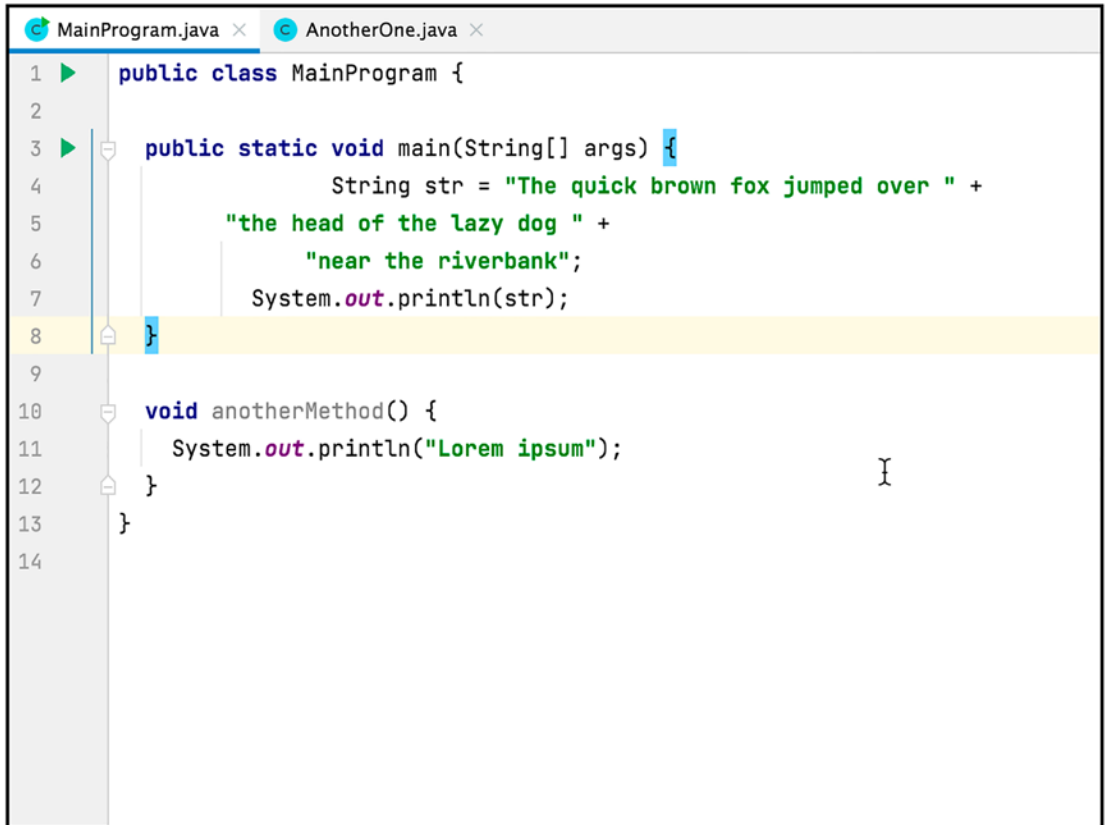

Figure B-4. Poorly formatted code

You can use the keys **Option + Command + L** (macOS) or **CTRL + Alt + L** (Linux, Windows) to quickly format the code. Figure B-5 shows the same code (as the one shown in Figure B-4), but this time, after applying the quick format.

5

```
  MainProgram.java ×     AnotherOne.java ×
 1 ▶      public class MainProgram {
 2
 3 ▶  ⊖     public static void main(String[] args) {
 4            String str = "The quick brown fox jumped over " +
 5                "the head of the lazy dog " +
 6                "near the riverbank";
 7            System.out.println(str);
 8  ⊖     }
 9
10  ⊖     void anotherMethod() {
11            System.out.println("Lorem ipsum");
12  ⊖     }
13      }
14
```

Figure B-5. *Properly formatted code*

Finding a word in the File

As well as using ⌘F, or **Ctrl F**, to find something, you can use ⌘G, or **F3**, to find the next occurrence of the word currently at the caret and move to it. Alternatively, use ^⌘G or **Ctrl**+**Alt**+**Shift**+**J** to select all instances of the current word.

I'm sure you already know how to find a word in a file. It's **Command + F** (macOS) or CTRL + F (Linux, Windows). Let's just add a little bit to your word-hunting arsenal. You can also use **Command + G** or **F3** to find the next occurrence of the word. Alternatively, you can use **CTRL + Command + G** or **CTRL + Alt + Shift + J** to select all instances of the current word.

Select and Surround

To add quotes around a word or words, highlight the text and type the quote. This also works for all types of brackets. The quotes or brackets will be placed directly outside the highlighted text.

Views

You can get more screen real estate by hiding all the tool windows individually. You can also use **Command + Shift + F12** (macOS) or **CTRL + Shift + F12** (Linux, Windows).
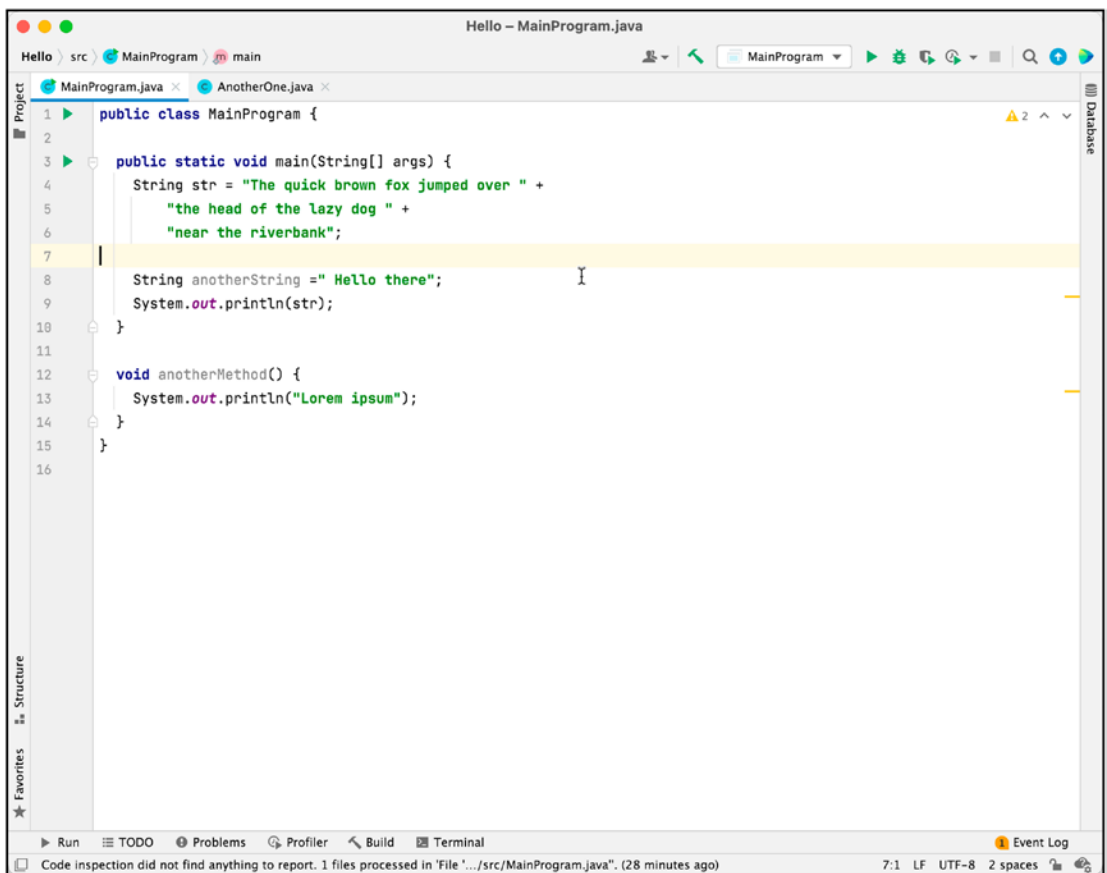
Figure B-6. *Maximized code editor view*

If you want to experiment a bit more on the available coding views, go to the main menu bar, then View ➤ Appearance (as shown in Figure B-7). Try out *Presentation, Distraction-Free, Fullscreen* or *Zen* mode.
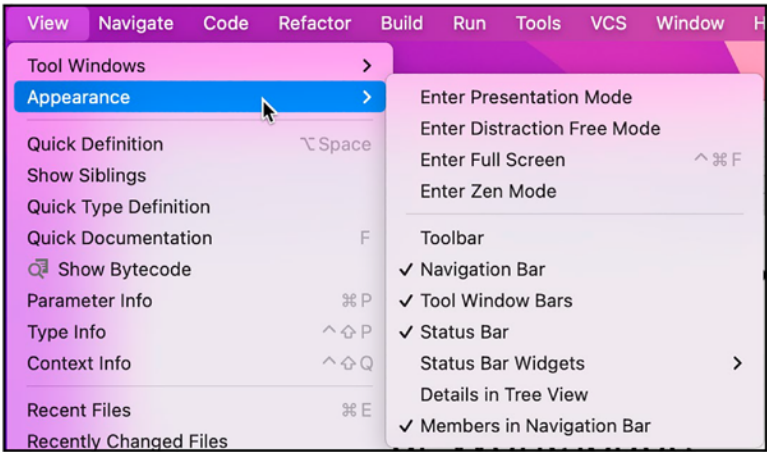
Figure B-7.   *View ➤ Appearance*

Key Takeaways

- IntelliJ is a chockful of goodies when it comes to coding assistance in. The Tip of the Day dialog is a good start in discovering these goodies. If you turned it off, you could always get to it by using the double Shift key, then type "tip of the day".

- IntelliJ favors the touch-typist, almost all of the actions you can do within the IDE has a keyboard shortcut. Learning these shortcuts will be time well-spent. You can look at the keymaps in the Preferences or Settings dialog — but a really quick way to get to it is double shift key, then type "keymap".