

M c l c

Mục lục	1
HƯỚNG DẪN SỬ DỤNG ECLIPSE	2
1 Các thao tác cơ bản	2
1.1 Create và import project.....	2
1.2 Build, run chương trình	2
1.3 Debug chương trình	3
1.4 Đóng gói chương trình.....	3
2 Một số thao tác khác	3
2.1 Tạo ra file Javadoc mô tả cho file source code.....	3
2.2 Chọn compiler tùy thích	5
2.3 Tùy chỉnh hiển thị LineNumber và lưu file UTF-8	5
2.4 Tùy chỉnh tự động chèn bracket và semicolon.....	6
2.5 Chèn thư viện trực tiếp và project	6
2.6 Sử dụng các UserLibrary	7
3 Các thao tác tăng tốc độ viết code	8
3.1 Sử dụng chức năng code template.....	8
3.2 Sử dụng chức năng quick fix.....	9
3.3 Các phím tắt thường dùng.....	10

HƯỚNG DẪN SỬ DỤNG ECLIPSE

1 Các thao tác cơ bản

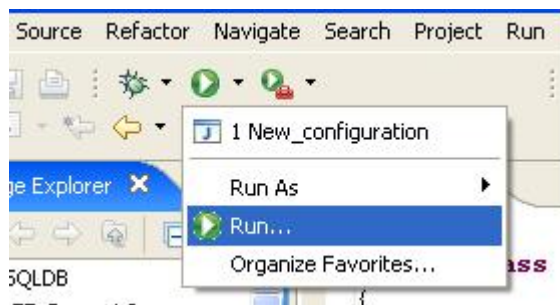
1.1 Create và import project

- Tạo project mới: Vào menu **File** chọn **New / Project**, chọn loại project mà bạn muốn tạo.

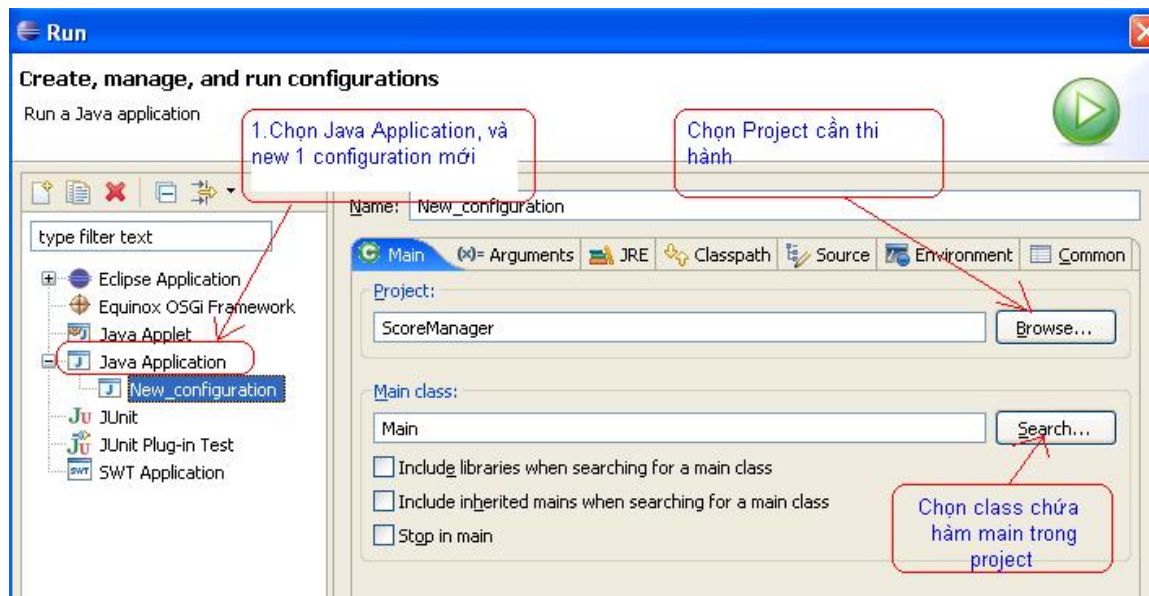
- Import project có sẵn: Vào menu **File** chọn **Import**. Trong hộp thoại **Import**, chọn mục **General / Existing project into workspace**, click **Next** tiếp theo để chọn project cần import, click **Finish**

1.2 Build, run chương trình

Click vào nút **Run** trên thanh công cụ, chọn **Run...**



Khi hộp thoại run mở ra ta chọn **New Java Application**. Sau khi chọn project xong thì chọn **Run** thì sẽ chạy chương trình.



1.3 Debug chương trình

Có thể debug chương trình ta cần đặt breakpoint tại một vị trí nào đó trong source code cần debug, rồi vào menu **Run** chọn **Debug...** (sau đó có thể hình ảnh trên). Lưu ý là có một số trình bảo vệ firewall (Bit Defender) khi đó chương trình sẽ không thể debug được.

1.4 Đóng gói chương trình

- Chọn menu project cần export trong cây explore rồi nhấn chuột phải chọn export.
- Trong cửa sổ Export ta chọn java/JAR file, chọn next.
- Trong cửa sổ Jar Export ta chọn đường dẫn cho file output.jar, chọn next.
- Tiếp tục chọn next, trong textbox MainClass ta cần chọn file class chứa hàm main trong chương trình của ta.
- Chọn Finish hoàn tất.

2 Một số thao tác khác

2.1 Tạo ra file Javadoc mô tả cho file source code

Khi viết các phương thức trong eclipse, nếu bạn tuân theo các quy tắc viết comment cho phương thức thì bạn sẽ có thể tạo ra các file doc mô tả về các phương thức, các lớp mà bạn viết, các mô tả này bao gồm mô tả về chức năng của hàm, các input, output, giá trị trả về và các Exception mà phương thức sẽ throws.

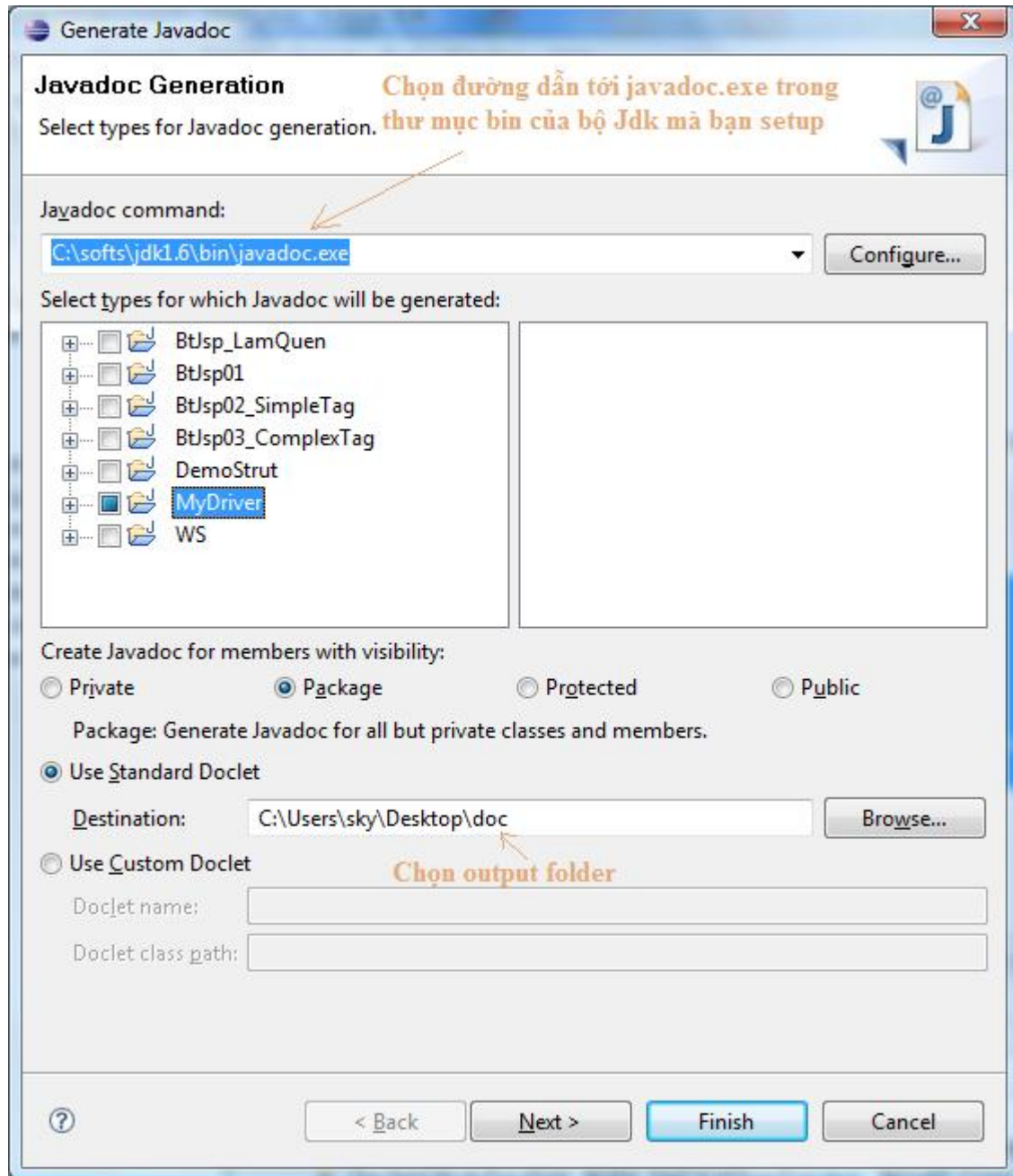
Để phát sinh tự động các comment đó ta cần chú ý vị trí tên hàm (public void myMethod(...)), nhấn tổ hợp phím Alt+Shift+J chương trình sẽ phát sinh tự động các comment đó. Vì còn lưu ý là chức năng viết các thông tin mô tả cần thiết. Lưu ý là nếu bạn viết các thông tin đó bằng tiếng Việt sẽ không gõ được Unikey thì cần phải sử dụng bảng mã Unicode để gõ tiếng Việt.

Đây là quy tắc comment:

1. /**
2. * Các mô tả cho phương thức
3. * @param iValue : mô tả cho parameter
4. * @param dValue : mô tả cho parameter
5. * @return mô tả giá trị trả về
6. */
7. public String myMethod(int iValue, double dValue) {
8. return "";
9. }

Sau khi tạo ra các comment ghi chú cho các phần theo quy định, bạn có thể tạo ra các file javadoc mô tả dự án mà bạn vừa viết.

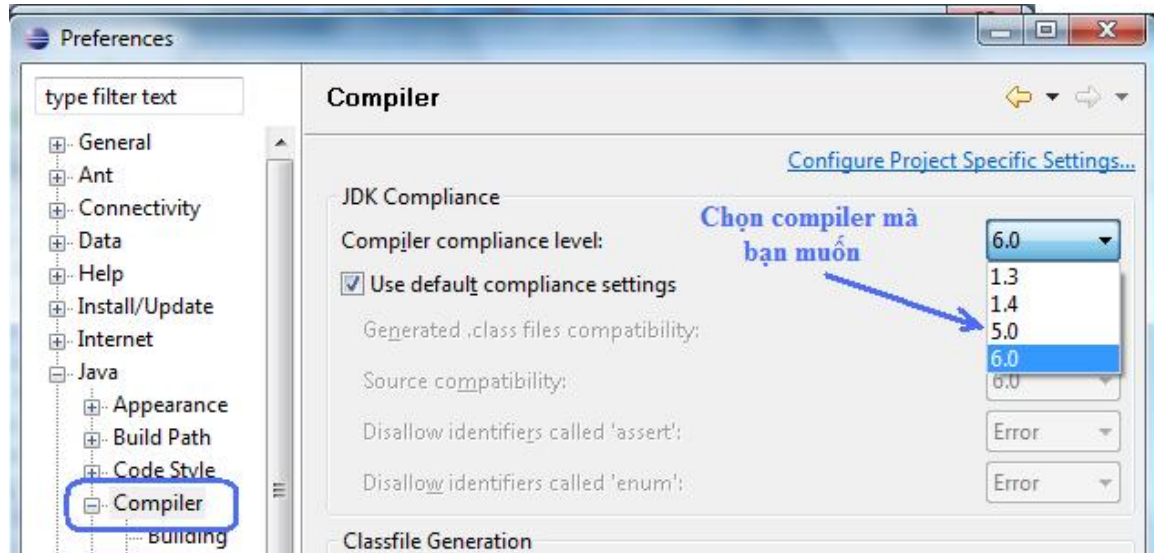
Vào menu project, chọn generate javadoc...



Bạn chọn **Next** tùy chọn thêm mô tả cấu hình khác rồi chọn thêm mô tả file doc của các thành viên mà bạn sử dụng; hoặc click **Finish** để tạo ra file javadoc.

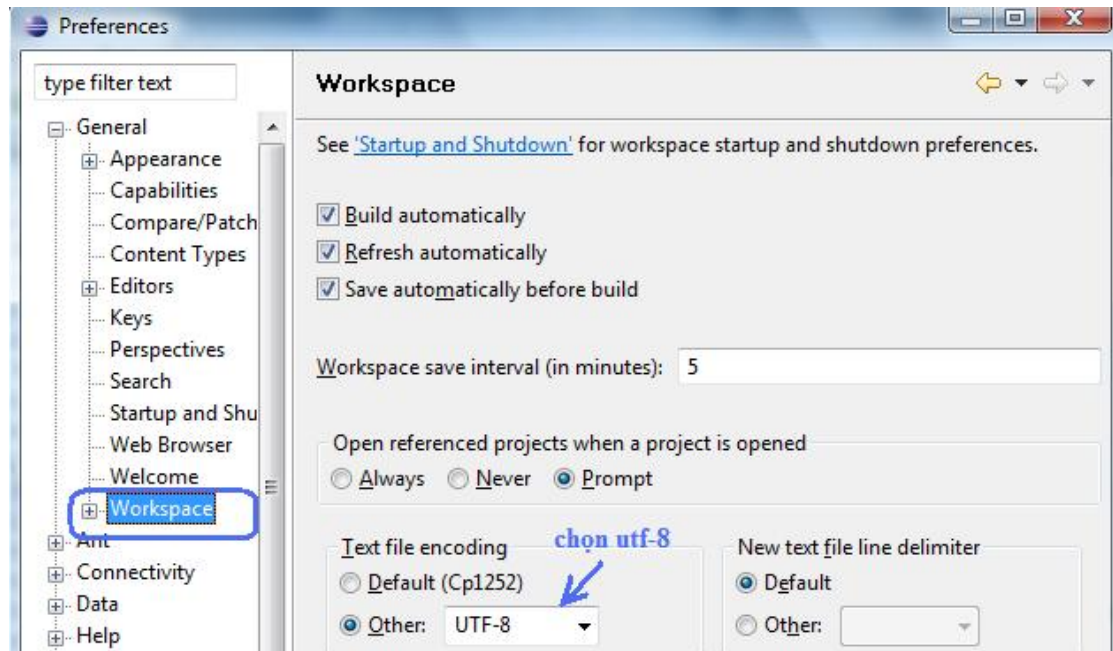
2.2 Chọn compiler tùy thích

Vào menu Window, chọn Preferences.



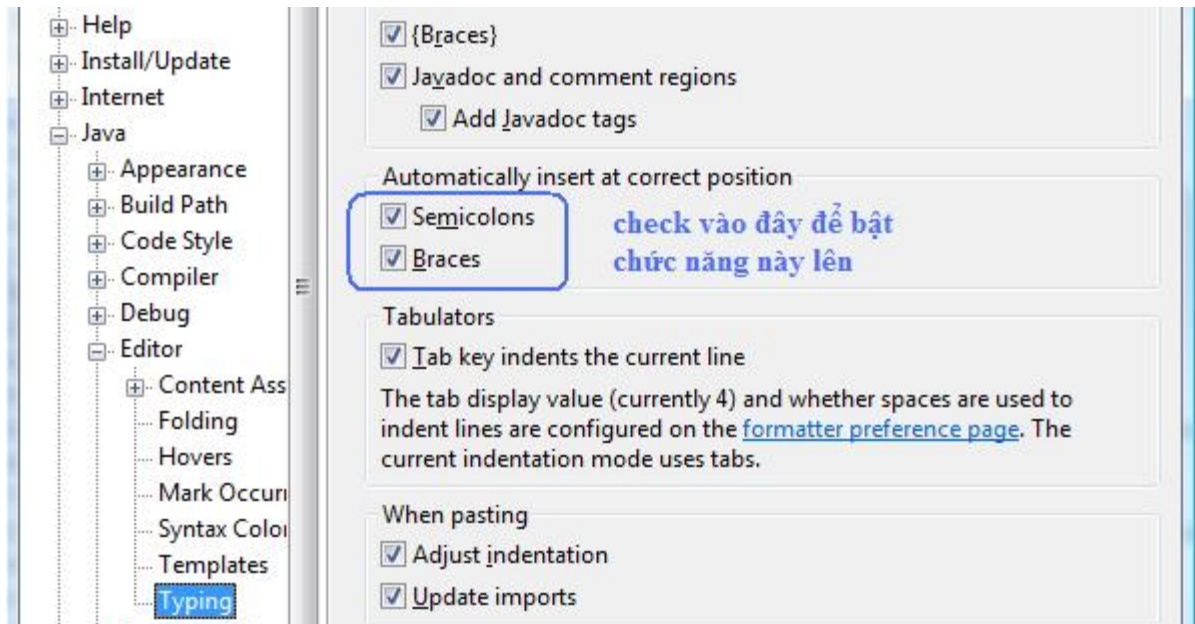
2.3 Tùy chỉnh hiển thị LineNumber và lưu file UTF-8

Hiển thị các dòng trong eclipse ta vào menu Preference. Mặc định eclipse sử dụng encode CP1252 nên ta không thể soạn thảo các ký tự tiếng Việt có dấu. Do vậy muốn lưu trữ file source theo định dạng UTF-8 ta làm như sau:



2.4 Tùy chỉnh việc chèn bracket và semicolon

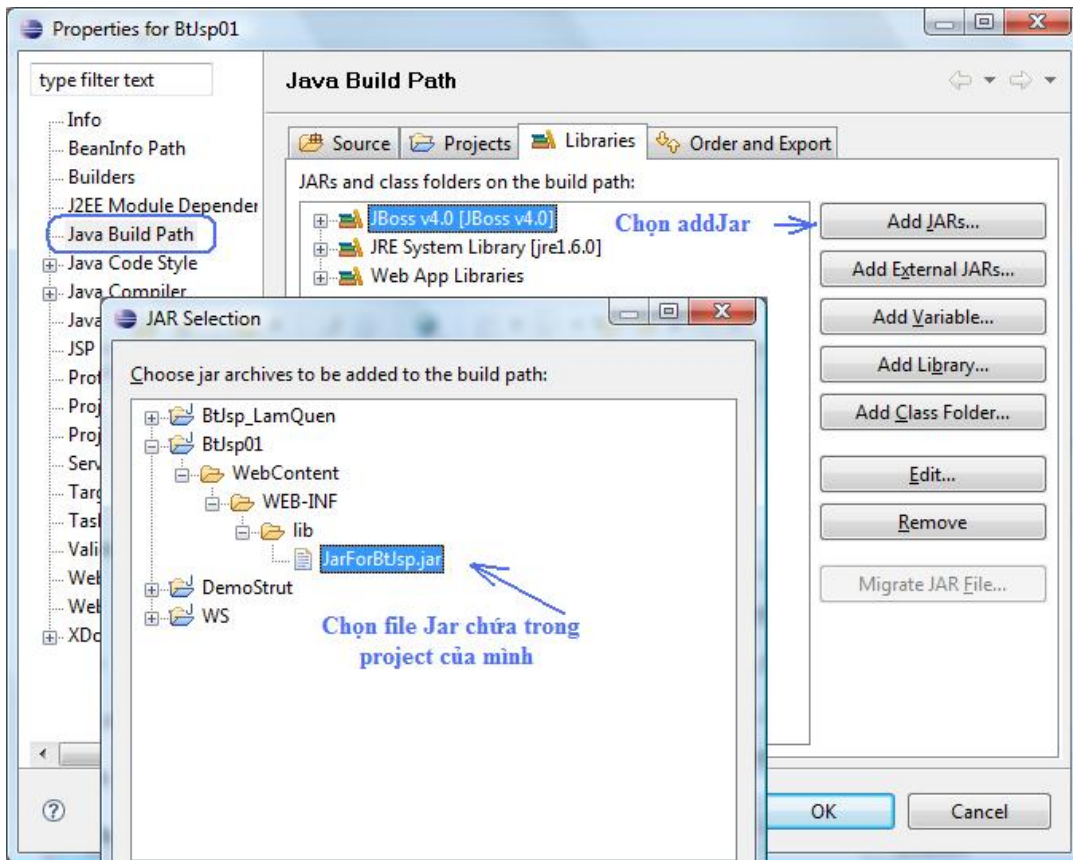
Trong khi code, bạn phải làm một số thao tác như di chuyển con trỏ xuống cuối dòng rồi ';' hoặc mở '{', như vậy thì thao tác code của bạn nào bị gián đoạn. Khi bạn chỉnh sửa này thì bạn có thể bật chức năng tự động chèn bracket và semicolon. Sau khi bật chức năng này trong khi code bạn có thể chèn ';' hoặc '{' bất cứ nơi nào trong dòng code.



2.5 Chèn thư viện class và project

Bạn có một thư viện class java, bạn tải về java library trên mạng, ... thì chúng ta cần nén trong file JAR, vậy là làm sao để đưa nó vào các project? Cách đơn giản nhất là bạn chỉ cần chép thư viện file thư viện đó vào project của mình và thì chỉ cần thêm một số thao tác Eclipse hiển thị các thư viện đó trong project của bạn.

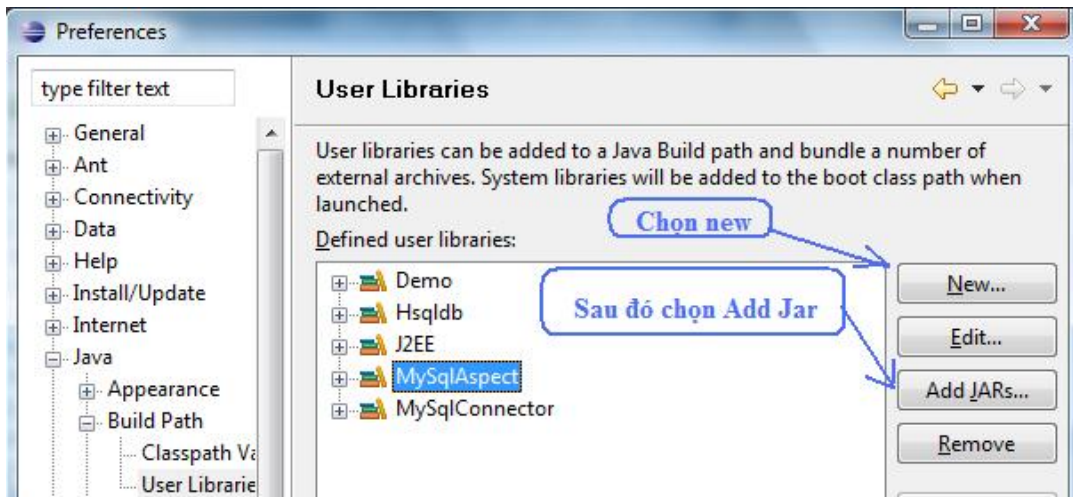
Sau khi đã chép thư viện vào, bạn chuột phải vào project đó và chọn properties:



Ghi chú: c u hình cái library s d ng trong project c l u trong file .classpath

2.6 S d ng các UserLibrary

B n có m t t p các th vi n th ng xuyên s d ng, m i l n sài b n ph i th c hi n các thao tác nh trên... th t m t th i gian, có cách nào kh c ph c vi c này hay không? Câu tr l i chính là s d ng UserLibrary. T o ra UserLibrary:



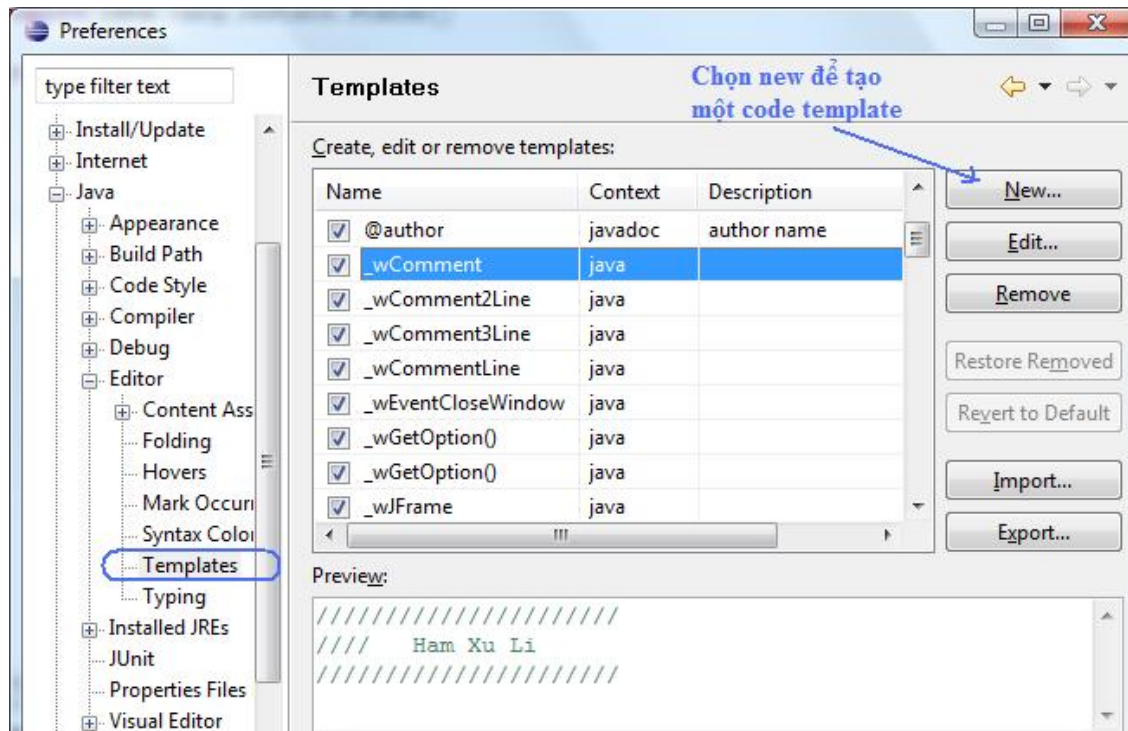
Add UserLibrary vào project: nh n chu t ph i vào project trong cây explore, ch n build path, user library; sau ó s m ra h p tho i Add Library (ho c vào properties c a project) ch n library c n add vào và click finish

3 Các thao tác t ng t c vi t code

3.1 S d ng ch c n ng code template

Trong khi l p trình có nh ng o n code b n ph i vi t nhi u l n, ho c có nh ng o n code quá dài, g i m b t th i gian vi t code eclipse cung c p cho ta m t ch c n ng r t m nh ó là code template.

T o code template:

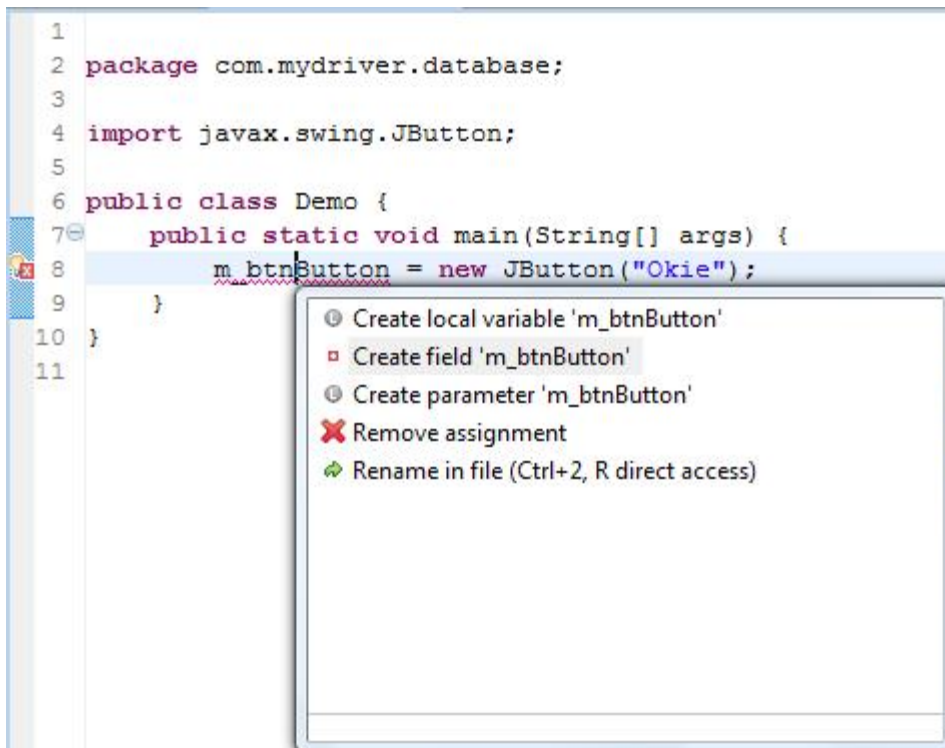


3.2 Sử dụng chức năng quick fix

Một chức năng khác rất مهم trong vi c h tr l p trình ó là ch c n ng quick fix. V i ch c n ng này b n có th ch s a t nhanh các l i trong khi l p trình. Ngoài ra nó còn giúp ta t o ra các o n l nh khai báo bi n, khai báo hàm, t o bi n input cho hàm, các local variable,... và còn r t nhi u cái khác.

g i ch c n ng quick fix ta t con tr chu t trên dòng ch a l i, nh n Ctrl+1 hi n ra khung ch nh s a.

Ví d v s d ng quick fix t o khai báo bi n.



3.3 Các phím tắt thường dùng

- Run: Ctrl+F11,
- Debug: F11, F6: step over, F7: step into, Ctrl+R: goto breakpoint
- Ctrl+PGUP, PGDOWN chuyển các tab
- Alt+'-' mở menu context của các TabEditor
- Ctrl+Shift+J tạo comment cho các hàm (comment có các annotation)
- Ctrl+O mở bảng goto method hoặc comment hàm
- Alt+Shift+Z bao bọc khối code bằng try, catch, while
- Ctrl+Shift+M import 1 package
- Ctrl+/- comment 1 dòng code
- Ctrl+Shift+P goto next bracket
- Ctrl+Alt+H tìm hàm nào gọi hàm tại vị trí chuột
- Alt+T - X exact method

ThS. Trần Tuấn - t-B môn CNPM-Vi n CNTT-TT

- Alt+T – S encapsulation field

- Ctrl+Q go to last edit