

# KỸ NĂNG LẬP TRÌNH

(*Phím tắt trong NetBeans*)

## 1/- Soạn thảo mã lệnh

Tên phím	Chức năng
Ctrl + E/X	xóa dòng tại con trỏ chuột
Ctrl + Shift + Up/Down	tạo bản sao 1 dòng
Ctrl + R	đổi tên nhiều biến cùng 1 lúc
Ctrl + Shift + O	tìm 1 file bất kỳ trong Project đang mở
Ctrl + Q	trở về vị trí edit cuối cùng
Ctrl + Shift + 1	Định vị 1 file trong project
Tab + Ctrl + 7	Mở nhanh Navigator
Ctrl + /	Bật tắt comment
Ctrl + B	Di chuyển đến chỗ khai báo function
Alt + Insert	sinh ra code
Ctrl + Shift + I	tự động thêm class cần thiết
Alt + Shift + F	định dạng code

## 2/- Biên dịch, kiểm tra và chạy chương trình

Tên phím	Chức năng
F9	biên dịch package/file
F11	build project chính
Alt + Shift + F	định dạng code
Shift + F11	xóa và build project chính
Shift + F6	chạy file hiện tại
F6	chạy project

Ctrl + F5	bắt đầu debugging project chính
Ctrl + Shift + F5	bắt đầu debugging file hiện tại
Ctrl + Shift + F6	bắt đầu debugging file
Ctrl + Shift + F8	đặt breakpoint mới

### 3/- Cấu trúc, lệnh, từ khóa

Tên phím	Cấu trúc lệnh, từ khóa
En	Enumeration
Ex	Exception
Ob	Object
Psf	public static final
Psfb	public static final boolean
Psfi	public static final int
Psfs	public static final String
St	String
ab	abstract
as	assert true;
bo	boolean
bcom	/**/
br	break;
ca	catch (
cl	class
cn	continue
db	double
df	default:

dowhile	do { } while (condition);
eq	equals
ex	extends
fa	false
fcom	//
fi	final
fl	float
forc	for (Iterator it = collection.iterator();it.hasNext();) {Object elem = (Object) it.next();}
fore	for (Object elem : iterable) {}
fori	for (int i = 0; i < arr.length; i++) {}
forl	for (int i = 0; i < lst.size(); i++) {
forst	for (StringTokenizer st = new
forv	for (int i = 0; i < vct.size(); i++) {
fy	finally{ }
ie	interface
ifelse	if (condition){}else { }
iff	if (exp) {}
im	implements
inst	if (exp instanceof Object) {
iof	instanceof
ir	import
le	length
na	native
newo	Object name = new Object(args);

pe	protected
pr	private
psf	private static final
psfb	private static final boolean
psfi	private static final int
psfs	private static final String
pst	printStackTrace();
psvm	public static void main(String[] args){}
pu	public
re	return
runn	Runnable runnable = new Runnable() {
serr	System.err.println (" ");
sh	short
sout	System.out.println (" ");
soutv	System.out.println("Object = " + Object);
st	static
su	super
sw	switch (var) { case val: break;
sy	synchronized
tds	Thread.dumpStack();
th	throws
tr	transient
trycatch	try {}
tw	throw
vo	volatile

