

ENCAPSULATION

PUBLIC → INTERFACE
QUELQUES METHODES BIEN CHOISIES

PRIVATE → CORPS
ATTRIBUTS
METHODES

 PACKAGES → **DEFAULT = PRIVATE** DS LE PACKAGE COURANT
DS LES AUTRES PACKAGE
DROIT ACCES FRIENDLY

ACCESSEURS = GETTERS & MANIPULATEURS = SETTERS

double getHauter() { return hauteur; }

void setHauter(**double** h) { hauteur = h; }

↓

PRIVATE double hauteur;
class Rectangle

(?) Pourquoi passer par un setter et non pas laisser l'attribut en public pour le modifier en faisant depuis le main ?

REPONSE

⚠ PAS DE SET POUR TOUT LES ATTRIBUTS ! (ENCAPSULATION)

LE SETTER PERMET :

- 1) MECANISMES DE VERIFICATION
- 2) GESTION DE L'ERREUR

③ LA MODULARISATION = MODIFIER UN PROG. DE FACON INDEPENDANTE

Si **PRIVATE double hauteur;** DEVIENS **PRIVATE int hauteur;**
CHANGEMENT TYPE ATTRIBUT

void setHauter(double h) { if () { } else hauteur = h; }

void setHauter(int h) { hauteur = h; }

MANIPULATION AU NIVEAU DE VOID UNIQUEMENT

SINON...

~~instance~~ ~~instance~~ ~~instance~~ ~~instance~~

TEST A REEcrire !

SHADOWING (ATTENTION!) = MASQUAGE

```
void setHauteur(double hauteur)
{
    hauteur = hauteur;
}
```

NON IDENTIQUES
A EVITER

①
MASQUAGE AVEC this.

```
void setHauteur(double hauteur) {
    this.hauteur = hauteur;
}
```

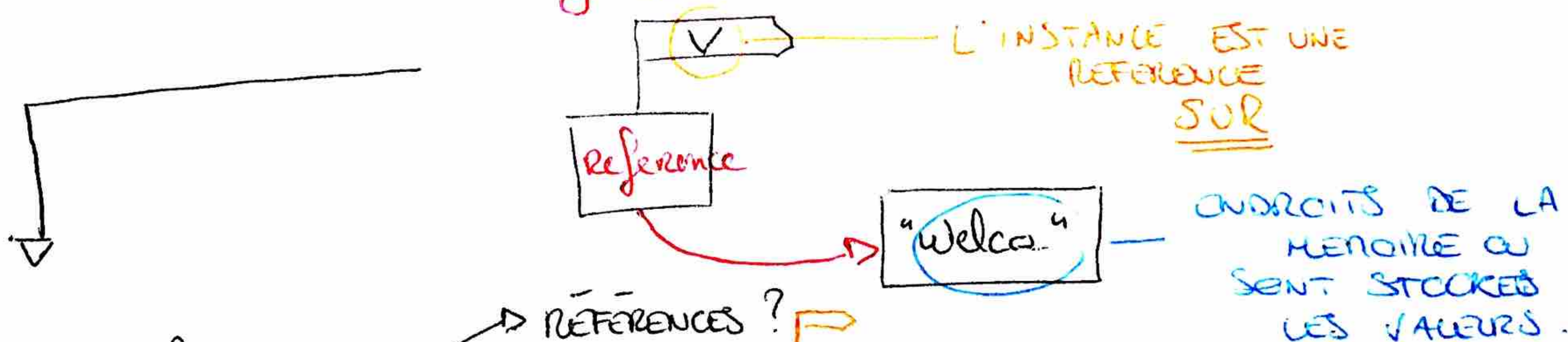
DESIGNE L'ATTRIBUT DE LA
CLASSE COURANTE

②
MASQUAGE AVEC
NON DIFFERENT

```
void setHauteur(double h)
    hauteur = hauteur;
```

OBJETS EN MEMOIRE

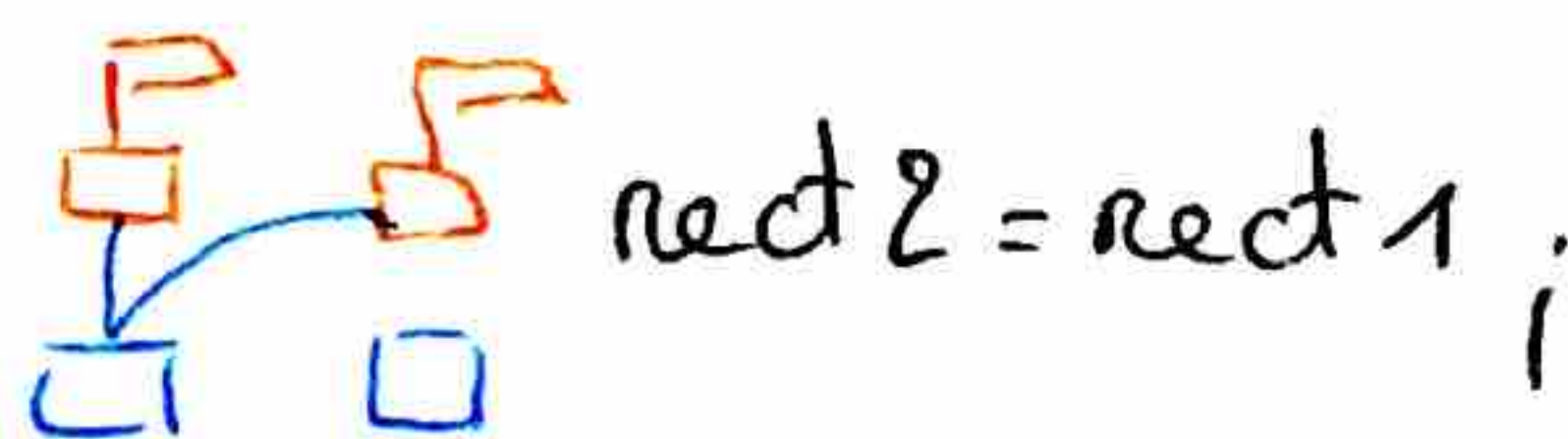
String v = "welco";



> COMPARER 2 OBJETS
egads.

REFERENCES? ☒
VALEURS? ☐

> AFFECTER 1 OBJET A
UN AUTRE
=



> AFFICHER 1 OBJET
toString.

REFERENCES? ☒ → sysout()
VALEURS? ☐ → sysout()

) avec toString()
dans la classe
de l'objet

CONSTANTE null

```
Rectangle rect = null;
```

↓
DECLARE UNE INSTANCE

POURQUOI FAIRE ?



↓
CONTROLLER SI LA REFERENCE POINTE VERS UN OBJET EN MEMOIRE

```
if (rect == null) { rect = new ... } (1) INSTANCIATION si null  
if (rect == null) { rect.surface(); } (2) APPEL DE METHODE si null
```

```
private final static Scanner CLAVIER = new Scanner(System.in);  
Rectangle rect = new Rectangle();  
double lu;  
Sysout("Quel hauteur");  
lu = CLAVIER.nextDouble();  
rect.setHauteur(lu);
```