

EX 1 :

1)

42

42

24

24

24

24

Fille a sa méthode donc quand on utilise printMeth ca imprime la méthode de fille.

2)

2 pour la fille . On peut y accéder en mettant meth() et super.meth()

1 seul dans le main : une meth() est dispo et ca depend on appelle avec quel objet. On y accède avec objetFille.meth()

3 et 4) Un champs est propre à l'instance, si un objet est instancié Mère, mere.meths donne la valeur du champs dans Mère, si un champs est instancié Fille, fille.meth retourne la valeur du champs dans fille.

Static

Champ

42

42

42

42

24

24

42

42

42

42

42

42

Ex2:

1) — —) quelques erreurs de compilation.

-On remarque que la fonction miage() n'est pas présente dans la classe mère

- Dans la classe mère la fonction h() renvoie un int tandis que dans la classe fille elle renvoie un char

-Dans la classe mère la fonction i() ne renvoie rien tandis que dans la classe fille elle renvoie un int

-Dans la classe fille le modificateur de la fonction e() est plus restreint que celui de la mère (alors que quand on redéfinit une méthode dans une classe fille il faut que le modificateur soit le moins restreint au maximum)

-Dans la classe fille les fonctions j(), k(),l(),m() lancent des exceptions différentes de chez la classe mère (Alors que quand on une exception IOException dans la classe mère on ne peut pas en avoir dans la classe fille car c'est plus général)

3)On parle de surcharge d'une méthode ou d'un constructeur lorsqu'il nous est permis de définir plusieurs fois une même méthode/constructeur avec des paramètres différents dans une sous classe sans changer la signature et les paramètres de la méthode mère.
On parle de redéfinition d'une méthode lorsqu'on réécrit une méthode de la classe mère dans la classe fille sans changer sa signature .

4)
mere.miage();// il ya une erreur de compilation
fille.miage();//"Miage"
mereFille.miage();//
((Fille)mereFille).miage();//"Miage"

mere.a();// la méthode est publique , donc on aura "Mere_a"
mereFille.a();// "Fille_a"
fille.a();// "Fille_a"
((Mere)mereFille).a();// "Mere_a"
mereFille.b(null);// Il y a une erreur de compilation car on ne peut pas mettre nul

mereFille.c();//Il y a une erreur de compilation car il y a nécessité d'un paramètre pour la fonction c.
mereFille.c(mere);// "Fille_c(Mere)"
mereFille.c(mereFille);// "Fille_c(Mere)"
mereFille.c(fille);//"Fille_c(Fille)"
fille.c(fille);//"Fille_c(Fille)"

mere.d();//"static Mere_d"
mereFille.d();//"static Mere_d"

mere.printF();//"Mere_f"
mereFille.printF();//"Fille_f"

mereFille.j();//"Fille_j"
mereFille.k();//"Fille_k"
mereFille.l();//"Fille_l"
mereFille.m();//"Fille_m"

EX3

Voir code