Annexe Transtypage en Kotlin

## De superclasse à sous-classe

|  |
| --- |
| JAVA |
| Soit un LinearLayout appelé linearLay contenant différents composants : |
| Button temp = (Button) linearLay.getChildAt(0); |
| ((Button)linearLay.getChildAt(3)).setText(“cliquez sur moi”); |
| if ( linearLay.getChild(2) instanceof TextField ) |
| KOTLIN |
| var temp:Button = linearLay.getChildAt(0) as Button |
| (linearLay.getChildAt(3) as Button).text=“cliquez sur moi”  Ou  (linearLay.getChildAt(3) as Button).setText(“cliquez sur moi”) |
| if ( linearLay.getChild(2) is TextField ) |

|  |
| --- |
| JAVA |
| Soit un ObjectOutputStream appelé oos contenant différents objets : |
| Maison m = ( Maison ) oos.readObject(); |
| System.out.println(((Maison)oos.readObject()).getNbChambres()); |
| Maison m = (Maison) oos.readObject();  Maison m2 = (Maison) oos.readObject(); |
| KOTLIN |
| var m = oos.readObject() as Maison |
| println (Cliquez ou appuyez ici pour entrer du texte.) |
| val m:Maison = oos.readObject() as Maison  val m2 = oos.readObject() as Maison |

|  |
| --- |
| **De type prédéfini à String** |
| JAVA |
| String s = String.valueOf ( 56 ); // s contient maintenant “56” |
| KOTLIN |
| var nombre = 56  var s = nombre.toString() // s contient maintenant “56” |
|  |
| JAVA |
| String usager = "Eric";  champTexte.setText("Salut" + usager ); |
| KOTLIN : On peut passer le contenu de variables sans avoir à concaténer de cette façon ( interpolation de strings ) : |
| var usager = "Eric"  champTexte.setText("salut **$**usager") |
|  |

|  |
| --- |
| **De String à type prédéfini** |
| JAVA |
| String nb = "52.95";  double nbEnDouble = Double.parseDouble(nb); // contient 52.95 |
| KOTLIN |
| var nb = "52.95"  var nbEnDouble = nb.toDouble() |
|  |
|  |