

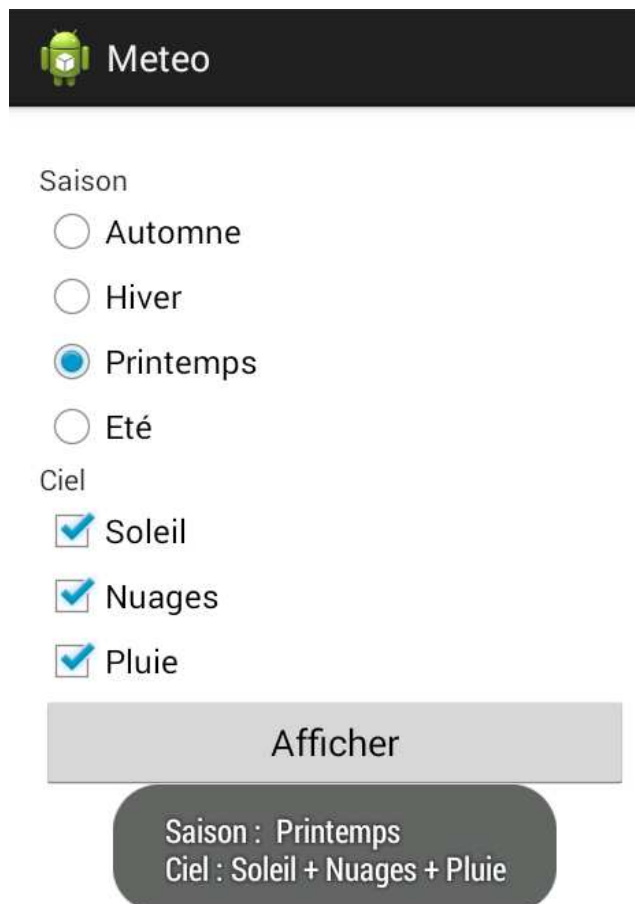
TP02

Matière : ATELIER DEVELOPPEMENT MOBILES 1

Classes : SEM21

L'objectif de ce TP est de programmer deux activités androïde. La première activité « Meteo » permet de cocher la saison et l'état du ciel puis d'afficher un message indiquant la saison et l'état du ciel. La deuxième activité « Couleurs » permet de choisir le coefficient de la transparence alpha (0..255), la valeur de la couleur rouge (0..255), la valeur de la couleur verte (0..255) et la valeur de la couleur bleue (0..255) puis d'afficher et d'appliquer la couleur résultante. Le TP permet l'utilisation de « RadioGroup », « RadioButton », « CheckBox » et « SeekBar ».

Exercice1 : Météo



Remarques pour « Meteo » :

- Si « Soleil » et « Nuages » ne sont pas cochés un message d'erreur s'affiche
- Si « Nuages » est décoché « Pluie » est désactivée et décoché
- Si « Nuages » est coché « Pluie » est activé.

- 1- Etape 1 : réalisation l'interface de l'activité
- 2- Etape 2 : modification du nom et de l'icône de lancement de l'application
- 3- Etape 3 : déclaration des constantes chaînes de caractères dans « strings.xml »
- 4- Etape4 : modification des propriétés des objets graphiques (text, layout_margin, id)
- 5- Etape5 : déclaration du viewBinding dans le gradle (dans android, dans buildTypes)

```
buildFeatures {  
    viewBinding = true  
}
```

- 6- Etape6 : codage de la classe MainActivity

```
private lateinit var amb : ActivityMainBinding  
  
override fun onCreate(savedInstanceState: Bundle?) {  
    super.onCreate(savedInstanceState)  
    enableEdgeToEdge()  
    initGraphique()  
}  
  
private fun initGraphique() {  
    amb = ActivityMainBinding.inflate(layoutInflater)  
    setContentView(amb.root)  
    ajouterEcouteurs()  
    actualiserPluie(amb.chNuages.isChecked)  
}  
  
private fun ajouterEcouteurs() {  
    amb.btnAfficher.setOnClickListener(object: View.OnClickListener{  
        override fun onClick(p0: View?) {  
            afficher()  
        }  
    })  
    amb.chNuages.setOnCheckedChangeListener {  
        object: CompoundButton.OnCheckedChangeListener{  
            override fun onCheckedChanged(p0: CompoundButton, p1: Boolean)  
                actualiserPluie(p1)  
        }  
    }  
}
```

```

private fun afficher() {
    if(amb.chSoleil.isChecked || amb.chNuages.isChecked) {
        var saison = "Saison: "
        saison += when (amb.rdgSaison.checkedRadioButtonId) {
            R.id.rdAutomne -> getString(R.string.automne)
            R.id.rdHivers -> getString(R.string.hivers)
            R.id.rdPrintemps -> getString(R.string.printemps)
            R.id.rdEte -> getString(R.string.ete)
            else -> "Inconnue"
        }
        var ciel = "Ciel: "
        ciel += if (amb.chSoleil.isChecked) "Soleil " else ""
        if (amb.chSoleil.isChecked)
            ciel += if (amb.chNuages.isChecked) "+ Nuages " else ""
        else
            ciel += if (amb.chNuages.isChecked) "Nuages " else ""
        ciel += if (amb.chPluie.isChecked) "+ Pluie " else ""
        Toast.makeText(this, "$saison \n $ciel", Toast.LENGTH_LONG).show()
    }else
        Toast.makeText(this, "Cochez Soleil ou Nuages SVP!",
            Toast.LENGTH_LONG).show()
}

private fun actualiserPluie(isNuages: Boolean) {
    if(isNuages){
        amb.chPluie.isEnabled=true
    }else{
        amb.chPluie.isEnabled=false
        amb.chPluie.isChecked=false
    }
}
}

```

Exercice2 : Couleurs



Remarques pour « Couleurs » :

- Si $\alpha > 0$ et $r > 0$ et $r = v$ et $b = 0$ alors écrire dans aperçu « Aperçu : Jaune »
- Si $\alpha > 0$ et $r > 0$ et $r = b$ et $v = 0$ alors écrire dans aperçu « Aperçu : Magenta »
- Si $\alpha > 0$ et $v > 0$ et $v = b$ et $r = 0$ alors écrire dans aperçu « Aperçu : Cyan »
- Utiliser `tvApercu.setBackgroundColor(Color.argb(a, r, v, b))`