

Consignes

Vous devez créer une application qui vous permettra de jouer au Démineur. L'application doit fonctionner en mode vertical seulement.

La grille de votre jeu doit être de 9 cases par 9 cases.

Chaque case sera gérée par un bouton que vous devez créer dynamiquement. (Ne créez pas les 81 boutons manuellement ... !)

Exemple avec un TableLayout (sera expliqué en classe) :

```
TableLayout table = (TableLayout) findViewById(R.id.tableLayout); TableRow
rangee = new TableRow(this);
```

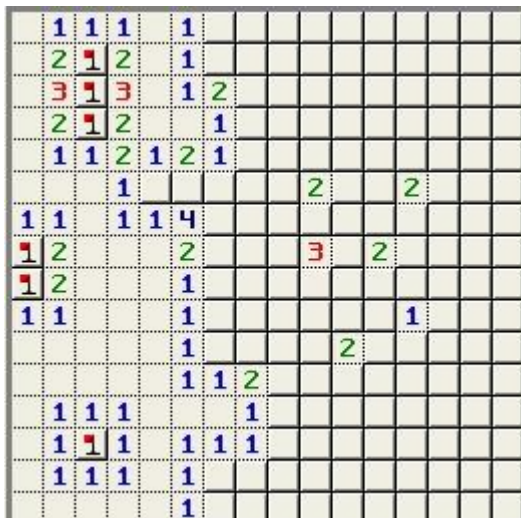
```
Button bouton = new Button(this);
```

```
table.addView(rangee);
rangee.addView(bouton, 100, 100);
```

Chaque case de la grille peut cacher une mine ou être libre. Le but du jeu est de découvrir toutes les cases libres sans faire exploser les mines, c'est-à-dire sans cliquer sur les cases qui les dissimulent.

Vous devez donc placer aléatoirement 10 mines à l'intérieur de votre grille. (Voir java.util.Random)

Chaque case qui ne contient pas de mine doit indiquer le nombre de mines adjacentes. C'est ce chiffre qui permet de déduire où sont placées les mines.



Vous devez mettre un bouton pour réinitialiser le jeu et un TextView pour montrer le nombre de mines restantes à trouver.

Les actions à gérer sur les boutons de la grille :

onClick :

- Si la case est une mine, on annonce la fin du jeu dans un Toast et on rend inactifs les boutons de la grille.
- Si la case est un chiffre, on affiche le chiffre dans le texte du bouton.
- Si la case est vide, vous pouvez seulement afficher un 0 ou une case grise. Un bonus sera accordé si vous concevez un algorithme qui permet de dévoiler toutes les cases vides adjacentes.

onLongClick :

- Vous devez placer un caractère dans le texte du bouton qui indique que vous pensez qu'il y a une mine sous cette case et vous devez décrémenter le nombre de mines dans le TextView.

Lorsque la partie est terminée, affichez un Toast et désactivez les boutons. Un clic sur le bouton de réinitialisation vous permettra de rejouer à nouveau.

Vous pouvez créer des classes supplémentaires qui vous permettront de mieux gérer votre grille.

Vous devez également limiter les avertissements d'Android Lint.

Faites attention aux commentaires, à la présentation et à la modularisation de votre code.

Votre application doit rouler dans l'émulateur Nexus 5 API 24 x86.

Remise

- Le travail doit se faire en équipe de deux.
- Vous devez remettre votre projet Android dans un fichier .zip sur Léa.
- Vous devez nettoyer votre projet (Clean project) avant de l'envoyer.
- La remise doit se faire avant mardi le 16 octobre 2018 à 23h59.