

Assignment 1 – Elon Sebbane 931176812

Exercise 1: Write your first Flutter app

Dry exercise

1. The two lines that are responsible to make the list infinitely scrolling are :

```
if (index >= _suggestions.length) {  
  _suggestions.addAll(generateWordPairs().take(10));  
}
```

If we delete these line we get an index error when we scroll to the bottom of the screen

2. We can use `ListView.separator`. The best way to construct a `ListView` if the list is finite and contains 100 items from the start is `ListView.builder` because the children are built lazy and can be infinite in number.
3. Calling `setState` notifies the framework that the internal state of this object has changed in a way that might impact the user interface in this subtree, which causes the framework to schedule a build for this State object. When we call to `onTap` handler, we need to update the lists so update the UI.

Exercise 2: Adding new features

Dry exercise

1. `MaterialApp` is a convenience widget that wraps a number of widgets that are commonly required for Material Design applications. It builds upon a `WidgetsApp` by adding material-design specific functionality, such as `AnimatedTheme` and `GridPaper`. The property `color` represents The primary color to use for the application in the operating system interface. The property `backButtonDispatcher` is a delegate that decide whether to handle the Android back button intent. The property `darkTheme` use the `ThemeData` argument when dark mode is required.
2. The key property of `Dismissible` controls how one widget replaces another widget in the tree. We need it in `Dismissible` since the widget can be removed from the widget list. If there are multiple dismissible widgets, make sure each has a unique key. Be careful not to use index as key as dismissing a widget can change the index of other widgets.