Quiz 18: State, State Management & The Provider Package/ Approach

1. What is "State" in a Flutter App?
    a. The UI your app renders over time.
    b. The data your app uses (and which may change over time) to render the UI.
    c. Data you download from web servers.

2. Why might StatefulWidget s alone not do the trick (in bigger/ more complex Flutter apps)?
    a. StatefulWidgets are not performant enough for bigger or more complex applications.
    b. StatefulWidgets can't be used for all kinds of state your app might depend on, hence certain features can't be implemented with StatefulWidgets alone.
    c. Multiple widgets may depend on the same state - sharing it via widget constructors can be cumbersome and inefficient (e.g. too many build() calls).

3. How does the Provider package/ the state management solution offered by the package work?
    a. You provide values (typically objects based on custom classes) in your widget tree. Whenever the data in your provided object changes, the entire widget tree (below the provider) will be rebuilt.
    b. You provide values (typically objects based on custom classes) in your widget tree. Further down the widget tree, you can then listen to changes in your provided objects. The widgets which are listening will rebuild upon such changes.
    c. You provide values (typically objects based on custom classes) in you the widgets which should update once the provided data changes.

Quiz 19: More on State Management & The Provider Package

Quiz Code Snippets

These code snippets belong to questions in the quiz. The question to which they belong will mention them. Generally, a snippet in the format "2x)" would belong to question 2 of the quiz

1a)

```
Widget build(BuildContext context) {
  return Column(children: [
    Text('This is a title!'),
    Consumer<SomeData>(builder: (ctx, someData, _) => Text(someData.someText)),
  ]);
}
```

1b)

```
Widget build(BuildContext context) {
   final someData = Provider.of<SomeData>(context);
   return Column(children: [
      Text('This is a title!'),
      Text(someData.someText),
   ]);
}
```

1. What's a key difference between using 1a) and 1b) in the attached text file?
    a. No big difference, just a different amount of code lines.
    b. The Consumer rebuilds the entire widget tree (i.e. re-runs the entire build() method), Provider.of() only runs a part of this widget's widget tree.
    c. **The second snippet rebuilds the entire widget tree (of this component), the first snippet only rebuilds the part inside of Consumer's builder method.**

2. What does Provider.of(context) NOT do?
    a. It does not re-run build() (of the widget it's being used in) if the data it's connect to (nearest SomeData object in this case) changes (via notifyListeners())
    b. **It does not provide data to other listeners (i.e. other Provider.of<...> calls in child widgets).**
    c. It does not react to every change in the nearest SomeData object (that is triggered via notifyListeners()).

3. What's a good example for local state which you would typically handle via a StatefulWidget instead of a provided object (i.e. instead of via the Provider package)?
    a. A list of products for your online shop app.
    b. Information about whether the user of the app is logged in or not.
    c. **A temporary selection on some input screen.**