a.

1. Mobile apps differ from desktop apps in how they are distributed and how they interact with user data on the device. Generally mobile apps are distributed through internet accessible App Stores controlled by the device manufacturer.  Apps can also be feature updated and security patched more easily through the App Store push mechanism.

b**.**

1. The iOS platform developed by Apple. However, the Android platform developed by Google.

2. The iOS vendor supported target programming languages are Objective C and the more recently introduced swift language. However, the Android vendor supported target languages are Java and more recently the Kotlin language.

3. Development for the iOS platform requires Apple hardware and their proprietary developer toolchain called Xcode. In contrast Google chose to license its  Android platform to third party OEM's to manufacture their own hardware with the option to customize individual Android distributions.

c.

1.**A pure native-app** is created using a language such as objective C or Java using native iOS or Android API's.The native app to be a mobile app and the native programming languages and interfaces are those chosen and developed by the platform vendors such as iOS and Android. The principal advantage of building apps natively is that the developer has full access to all the features. Native apps can often be first to take advantage of new platform features or bug fixes and enjoy faster approval into app stores for distribution to their end users. The key disadvantage of developing natively is the relative complexity of the native environments for developers.

2. **Natively wrapped-apps** use WC3 web standards. A natively wrapped-App is a web-app which has been packaged up as a native app and is distributable through the App Store mechanism. Additionally natively wrapped-apps have much better access to hardware features than pure web-apps. Web-apps generally perform less well and are less responsive than native-apps, and this comes about for the same two reasons as for web apps.

d.

1. The native web-app approach brings together the best ideas from each of the alternatives to offer the most portable and most capable app development model in use today.

2. A native web-app can be distributed and executed like normal native-apps.

3. The approach of native-web apps offers the possibility of greater productivity, portability, faster development and easier maintenance compared with the pure native-app approach.

e.

Similarities :

1.   They are created using web technologies like javascript and CSS.

2.   They can be stored on the phone for offline usage.

3.   Typically they can be aware of and adapt to the fact that the screen is relatively small and organizes the user interface elements accordingly.

Differences :

1.    The main difference is that there is no document object model or DOM in native web apps in the way that you would find in a web-app.

2.    Native web-app API's are not completely portable between target platforms for example iOS and Android, however web-apps are portable across all mobile platforms including iOS and Android and also desktop platforms such as Windows and Mac.

3.    The W3C suite of web-app is simpler and more intuitive set of languages and API's for use by developers. But native web-app API's are much closer to native API, so they allow the developer to leverage platform specific functionality and behavior that is not easily achieved with pure web-apps.

4.    In fact a web app runs in the browser on any environment. However, a native-app has no browser involved in either the development or the runtime of these apps.