
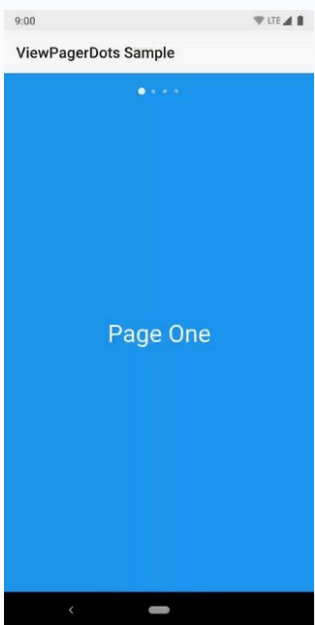
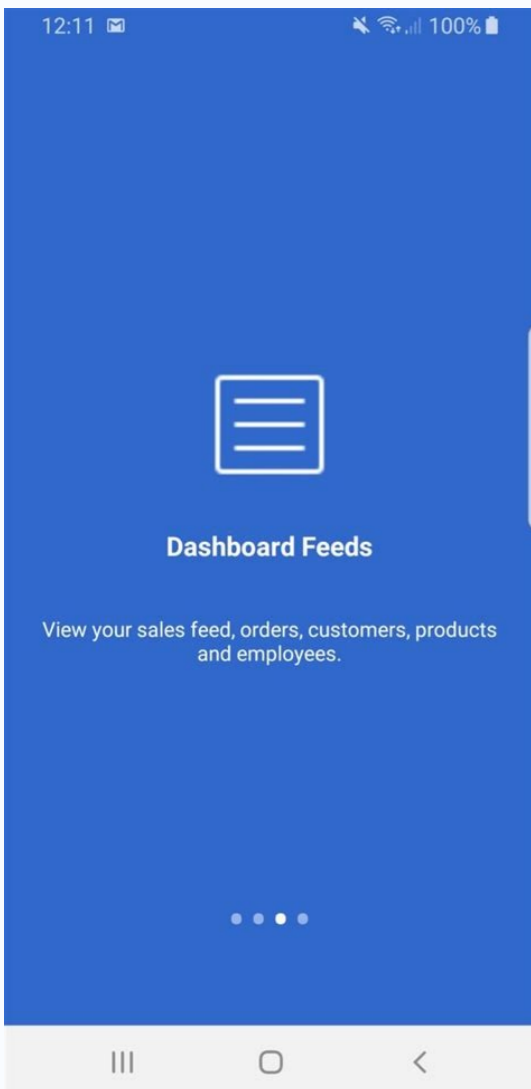
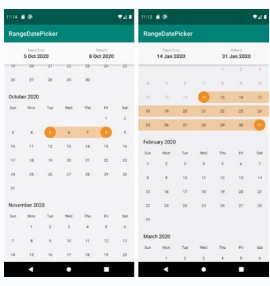
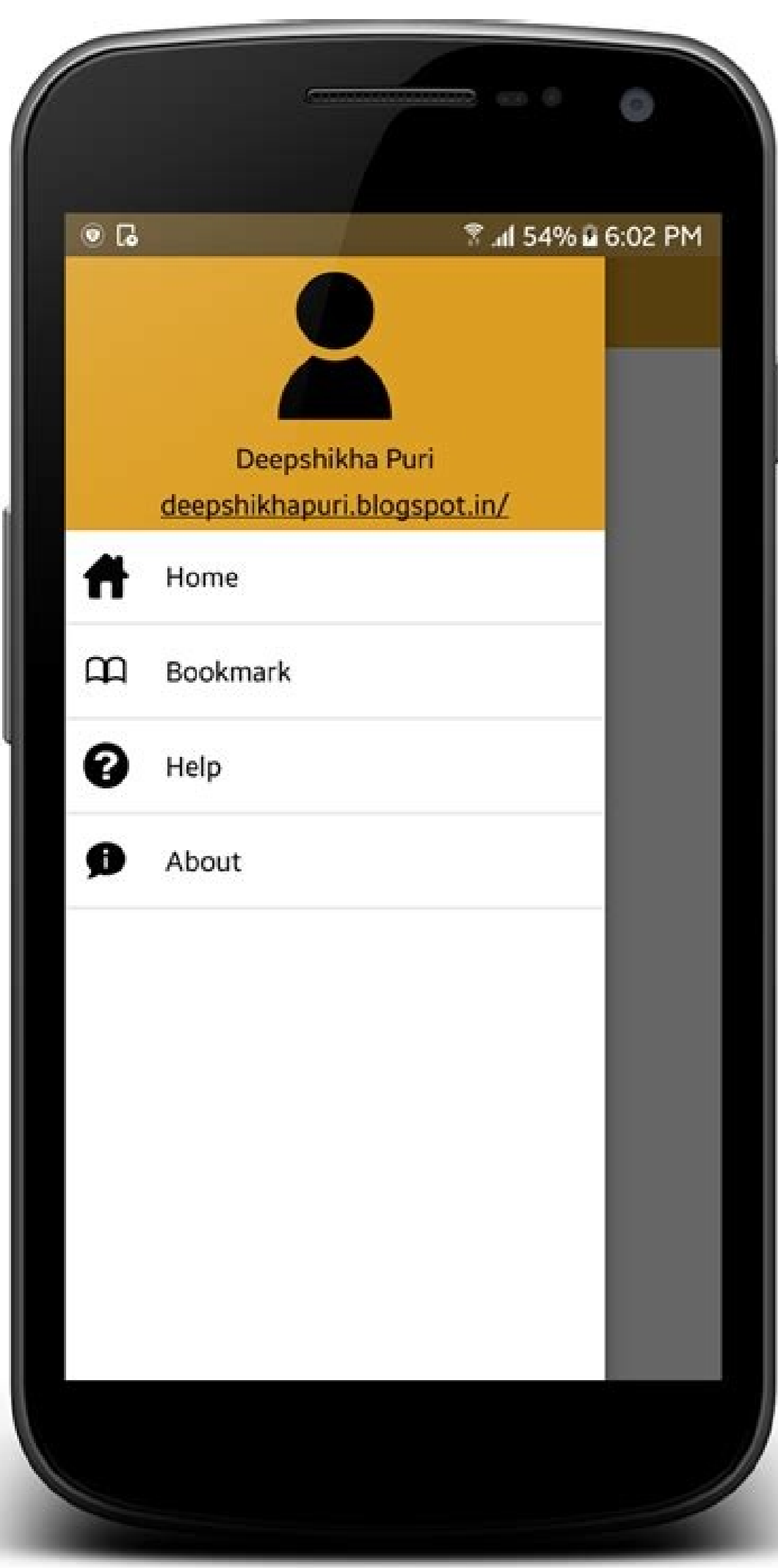


Kotlin navigation drawer

 I'm not robot  reCAPTCHA

Continue



The navigation belt is the belt of the user interface that displays the main navigation menu of the application. This is the most common function provided by Android. It is also an important function of the user interface that offers actions preferred by users, such as the edition of user profiles, the modification of the applications settings, etc. In this tutorial we will see how to create an Android navigation bar. The navigation belt appears when the user moves the finger from the left edge of the operation. They can also find it by touching the application icon on the home activity bar. Let's start -> Create a menu folder in the Res. To implement the menu. Step 2: Create a system for our navigation (optional) header. I want to add the title to my navigation drawer as follows: Step 3: Working with activity_main.xml. To adapt the basic elements needed to the navigation drawer, change the upper element of bond-> drawerlayout. Perform the following code in activity_main.xml. Step 4: Work with MainActivity.kt. To view the menu icon on the action bar and implement the opening/closing function of the navigation belt, follow the following code in the MainActivity.kt file. Ready -> Source Github: You can also watch my movie in this tutorial on YouTube: thanks for reading and good programming! Over 820 pages in the last chapter of this series of chapters dedicated to the design components of the Android equipment will be discussed by the topic of the navigation drawer. Together with the DrawerLayout classes, NavigationView and ActionBarDrawerToggle, the navigation drawer has the shape of a belt that appears on the left edge of the screen after the user selects and contains a series of options and subtitles, which can be selected to perform tasks on the screen and contains a series of options chosen by the user, generally designed to facilitate navigation in another part of the application. In Fig. For example, Figure 47-1 shows an integrated navigation belt in the application of Google Play: the navigation belt is made up of a navigation bar is a UI bar that displays the main navigation menu of an application and is the most common feature provided by Android. It is also a basic user interface feature that provides user-desired actions such as changing user profiles, changing app settings, etc. In this tutorial, we will discuss how to create navigation bars on Android. The navigation bar appears when the user moves their finger to the left edge of the activity when the navigation bar appears. They can also find it by tapping the app icon on the home taskbar. Let's start -> Create a menu in the RES folder. To implement the Navigation Drawer menu, change the parent using ConstraintLayout->StrockerLayout. Run the following code in Active_main.xml. Step 4: Working with MainActivity.kt file. To display the menu icon in the activity bar and give you an idea of the functionality of the Open Lock navigation drawers, run the following code in your MainActivity. With Android Studio 3.0/Android 8 Edition Book. Get the fully updated Android Studio Electric Eel Kotlin Edition System eBook (\$29.99) Edition Format android Studio Electric Eel - Kotlin eBook (PDF/EPUB, Kindle) 820+ pages, Kotlin eel. The final chapter on Android Material Design components will cover the topic of navigation drawers. The navigation drawer, which contains the DrawerLayout, NavigationView, and ActionBarDrawerToggle classes, takes the form of a bar that appears on the left side of the screen when the user selects it, and contains several options and child elements that can be selected at launch, time, tasks. For example, Figure 47-1. In the middle, Applockos shows the navigation drawer built into the Google Play app: navigation drawer sets. The following components: An example of the navigation view component embedded as a child of the drawer layout. A menu resource file containing options to display in the navigation bar. An optional layout resource file containing the content displayed in the head area of the navigation area. A listener assigned to the navigation view to recognize when the user has selected an item. Example of ActionBarDrawerToggle to connect and sync navigation bar with app bar. ActionBarDrawerToggle also shows a drawer screen on the taskbar, which shows the drawer when typed. The XML listing below shows an example implementation of the navigation drawer, which includes a DIRECTION instructions for inclusion for a second layout file containing the standard layout of the application strip. DrawerLayout XMLNS: android = " xmlns: app = " xmlns: tools = "http : // schemas .android.com/tools "android: id = " @+id/drawer layout "android: layout width = " Match parent "android: layout height = " Match parent "Android: fitsSystemWindows = " Tool "Tools: opendRwer = " Launch " -> report this screen. Tap the screen with the drawer in the apps multifunction bar, the drawer will appear automatically. Whether the draw is currently open can be determined by calling the isDrawerOpen() method of the DrawerLayout object and displaying the gravity setting: isDrawerOpen() (GravityCompat.start) (// Drawer Open) GravityCompat setting. Start specifies the drawer that opens along the x-axis of the layout. An open drawer can be concluded by calling the close() method. Conversely, a drawer can be opened using the OpenDrawer() method: Drawer.openDrawer(GravityCompat.start) EbookFrenzy.com. You have read an example from Android Studio 3.0 / Android 8 books. Publish this post as an eBook (\$29.99) Android Studio Electric Exotics -kotlin Edition eBook (PDF/EPUB, Kindle) contains 93 chapters and more than 820 pages selection processing in the navigation drawer. The first step is to specify an object that will act as an element selection listener. This is accomplished by calling the setNavigationItemSelectedListener() method by referencing the NavigationView instance in the layout and referencing the object that acts as a listener. Usually the listener is configured as the current action, for example navigationView.setNavigationItemSelectedListener(this) The second step is to introduce onNavigationItemSelectedListener() to the Lister. This method is called whenever a selection in the navigation drawer is selected and names the selected menu item as an argument, which can then be used to get and define the selected item: fun onNavigationItemSelectedListener (MenItem): Bow { A // use navigation object Click here . You have read one chapter of eBookFrenzy.com Android Studio 3.0 / Android 8. Electric Oel Essentials - Kotlin Ebook Edition (PDF/EPUB, Kindle), 93 chapters and more than 820 pages, if they are useful and as explained in the example above . it is also important to conclude. The navigation drawer can be loaded from the po drawer in any action, when creating a new project or adding a new enterprise to an existing project, access to the selection of the navigation drawer activity is to create all components and requirements needed to implement the navigation drawer, requires adjusting the default settings. If needed. Using the Create New Project for Android Studio by entering the NAVDRAWERDEMO field in the application name and ebookfrenzy.com as setting up the company domain before clicking the next button. Continue on the screens by requiring the creation of the Navdraweractivity Navigation Panel activity with the appropriate Action_nav_drawer settings file. Click Finish to start the project creation process. Source files Model layout This is the source file of the highest level layout. It has a drawer container and a subordinate navigation view. The NavigationView declaration in this file states that the outlet layout of the socket is included in the NAV_HEADER_NAV_DRAWER.xml file and the setting menu is in the ACTIVATION_NAV_DRAWER_DRAWER.xml file. In addition, it contains a link to app_bar_nav_drawer.xml file.app_bar_nav_drawer.xml. This resource layout is part of the ACTIVATION_NAV_DRAWER.xml file and is a standard file layout file created in the coordinator. chapters. As in previous examples, this file contains a directive for the inclusion of a content file, which in this case is called content_nav_drawer.xml.Content_nav_drawer.xml. Default layout of the content of the content of the activity layout. This layout consists of restrictions to container and Hello World! Texview.nav Header nav_drawer.xml - which refers to the NavigationView element in the Action_nav_drawer.xml file This is the location of the headligator in the socket layout. Layouts files, side_nav_bar.xml file located under App -> Draw can edit and change the colors used in the socket header. By default, this file declares a rectangular color transition that moves horizontally from dark to light. Read the sample chapter from Android Studio 3.0/Android 8 Edition. The release has 93 chapters and more than 820 pages, the menu options shown in the navigation drawer can be found in the Activity_NAV_DRAWER_DRAWER.XML file in the Application -> RES -> -tool Project window. By default, the menu consists of a series of text captions with accompanying characters (files each located in the draw folder). For more information on menu source files, see Creating and Managing Overflow Menus and Management on Androidbookfrenzy.com. Electric Eel Studio Kotlin Edition about this eBook publication (\$29.99) Format android Studio Electric Eele Essentials - Kotlin Edition eBook (PDF/ePub, Kindle) The output contains 93 chapters and more than 820 pages template method - Codthe oncreate () located at Navigractivity.kt. Kt. The file does most of the initialization work required for the navigation socket: val "Flip", a 6 a a in which it is related to the current activity, a drawerLayout object was initialized, which in the application contained a toolbar and two strings of strings, opening and closing drawers, closing act. Description ions for access purposes. The Barroggle event object is then assigned and synchronized as a box listener. The code then obtains a reference to the NavigationView instance before selecting the current activity as the receiver for all items created in the navigation drawer. The current activity is currently declared as boxes, the AnnavigationItemselected() method is also implemented in the NavRowerActivity.kt file. The implementation of this method is in the steps described earlier in this chapter has been added to Android Studio. This method is added for situations where an activity has a back button to return to the previous activity screen. The code in this method ensures that the drawer is closed until the application returns to the previous activity screen: override funky onbackessed () a a » . a a » a a » a a » User. Drawers contain menu options that can be selected and serve as a useful navigation tool within the Material Design guidelines. While any activity can add a navigation strip, the quickest way is to use the navigation strips activity template from Android Studio and customize it to fit your specific needs. This section describes the components that make up the navigation bar and highlights how they are implemented in Template.bookfrenzy.com. You are reading a chapter of Android Studio 3.0 / Android 8 Edition Android Studio 3.0 / Android. Ebook Election Eel Ensentions-Kotlin Edition Ebook (PDF/EPUB, Kindle) Election Elections-Kotlin (PDF/EPUB, Kindle) e-Book The book edition has 93 chapters and more than 820 pages, more than 820 pages

