

Spis treści

Wprowadzenie

Autorzy książki Android. Programowanie aplikacji. Rusz głową! (iv)

Dla kogo jest ta książka? (xxx)

Wiemy, co sobie myślisz (xxxi)

Wiemy, co sobie myśli Twój mózg (xxxi)

Metapoznanie - myślenie o myśleniu (xxxiii)

Oto co MY zrobiliśmy (xxxiv)

Przeczytaj to (xxxvi)

Zespół recenzentów technicznych (xxxviii)

Podziękowania (xxxix)

ROZDZIAŁ 1. Zaczynamy. Skok na głęboką wodę

Witamy w Androidowie (2)

Platforma Android w szczegółach (3)

Oto co mamy zamiar zrobić (4)

Środowisko programistyczne (5)

Zainstaluj Android Studio (6)

Stwórzmy prostą aplikację (7)

Jak stworzyć aplikację? (8)

Aktywności i układy z wysokości 15 tysięcy metrów (12)

Jak stworzyć aplikację? (ciąg dalszy) (13)

Właśnie utworzyłeś swoją pierwszą aplikację na Androida (15)

Android Studio utworzy pełną strukturę katalogów aplikacji (16)

Przydatne pliki projektu (17)

Edycja kodu z użyciem edytorów Android Studio (18)

Uruchamianie aplikacji w emulatorze Androida (23)

Tworzenie wirtualnego urządzenia z Androidem (24)

Uruchamianie aplikacji w emulatorze (27)

Postępy możesz obserwować w konsoli (28)

Ale co się właściwie stało? (30)

Usprawnienie aplikacji (31)

Czym jest układ? (32)

Plik activity_main.xml zawiera dwa elementy (33)

Aktualizacja tekstu wyświetlanego w układzie (34)

Weź aplikację na jazdę próbną (35)

Twój przybornik do Androida (36)

ROZDZIAŁ 2. Tworzenie interaktywnych aplikacji. Aplikacje, które coś robią

W tym rozdziale napiszemy aplikację Doradca piwny (38)

Utworzenie projektu (40)

Utworzyliśmy domyślną aktywność i układ (41)

Dokładniejsza prezentacja edytora projektu (42)

Dodawanie przycisku w edytorze projektu (43)

Plik activity_find_beer.xml zawiera nowy przycisk (44)

Dokładniejszy przegląd kodu układu (45)

Weź swoją aplikację na jazdę próbną (49)

Podawanie tekstów na stałe utrudnia lokalizację (50)

Utworzenie zasobu łańcuchowego (51)

Zastosowanie zasobu łańcuchowego w układzie (52)

Kod pliku activity_find_beer.xml (53)

Dodawanie wartości do komponentu Spinner (56)

Dodanie elementu string-array do pliku strings.xml (57)

Jazda próbna komponentu Spinner (58)

Musimy zadbać o to, by przycisk coś robił (59)

Niech przycisk wywołuje metodę (60)

Jak wygląda kod aktywności (61)

Dodaj do aktywności metodę onClickFindBeer() (62)

Metoda onClickFindBeer() musi coś robić (63)

Dysponując obiektem View, można odwoływać się do jego metod (64)

Aktualizacja kodu aktywności (65)

Pierwsza wersja aktywności (67)

Co ten kod robi? (68)

Tworzenie własnej klasy Javy (70)

Co się dzieje podczas wykonywania tego kodu? (74)

Jazda próbna - test aplikacji (75)

Twój przybornik do Androida (76)

ROZDZIAŁ 3. Wiele aktywności i intencji. Jakie są Twoje intencje?

Aplikacja może zawierać więcej niż jedną aktywność (78)

Oto struktura naszej aplikacji (79)

Zaczynamy: utworzenie projektu (79)

Aktualizacja układu (80)

Utworzenie drugiej aktywności i układu (82)

Przedstawiamy plik manifestu aplikacji na Androida (84)

Intencja jest rodzajem komunikatu (86)

Co się dzieje po uruchomieniu aplikacji? (88)

Przekazanie tekstu do drugiej aktywności (90)

Aktualizacja właściwości widoku tekstowego (91)

Metoda putExtra() zapisuje w intencji dodatkowe informacje (92)

Aktualizacja kodu aktywności CreateMessageActivity (95)

Zastosowanie informacji przekazanych w intencji w klasie ReceiveMessageActivity (96)

Co się dzieje, gdy użytkownik kliknie przycisk Wyślij wiadomość (97)

Możemy zmienić aplikację tak, by wiadomość była wysyłana do innych osób (98)

Jak działają aplikacje na Androida (99)

Utworzenie intencji określającej akcję (101)

Zmiana intencji w celu użycia akcji (102)

Jak Android korzysta z filtrów intencji? (106)

A co, jeśli chcemy, by użytkownik ZAWSZE wybierał aktywność? (112)

Co się dzieje w momencie wywoływania metody createChooser()? (113)

Zmień kod, by wyświetlać okno dialogowe (115)

Twój przybornik do Androida (118)

ROZDZIAŁ 4. Cykl życia aktywności. Była sobie aktywność

Jak właściwie działają aktywności? (120)

Aplikacja stopera (122)

Dodanie zasobów łańcuchowych (123)

Jak będzie działał kod aktywności? (125)

Działanie kodu obsługującego przyciski (126)

Metoda runTimer() (127)

Pełny kod metody runTimer() (129)

Kompletny kod aktywności StopwatchActivity (130)

Obrót ekranu zmienia konfigurację urządzenia (136)

Stany aktywności (137)

Cykl życia aktywności: od utworzenia do usunięcia (138)

Zaktualizowany kod aktywności StopwatchActivity (142)

Co się stanie po uruchomieniu aplikacji? (143)

Tworzenie i usuwanie to nie cały cykl życia aktywności (146)

Zaktualizowany kod aktywności StopwatchActivity (151)

Co się dzieje podczas działania aplikacji? (152)

A co się dzieje, jeśli aplikacja jest tylko częściowo widoczna? (154)

Cykl życia aktywności: życie na pierwszym planie (155)

Zatrzymanie stopera w razie wstrzymania aktywności (158)

Implementacja metod onPause() oraz onResume() (159)

Kompletny kod aktywności (160)

Co się stanie po uruchomieniu aplikacji? (163)

Wygodny przewodnik po metodach cyklu życia aktywności (167)

Twój przybornik do Androida (168)

ROZDZIAŁ 5. Widoki i grupy widoków. Podziwiał widoki

Interfejs użytkownika aplikacji składa się z układów i komponentów GUI (170)

Układ LinearLayout wyświetla widoki w jednym wierszu lub w jednej kolumnie (171)

Dodawanie pliku zasobów wymiaru w celu zapewnienia spójnych wypełnień w układach (174)

Stosowanie marginesów do oddalania widoków od siebie (176)

Zmieńmy nieco prosty układ liniowy (177)

Rozciągaaaaamy widok, zwiększając jego wagę (179)

Wartości atrybutu android:gravity (183)

Kompletny układ liniowy (186)

Układy FrameLayout rozmieszczają widoki jeden na drugim (188)

Dodanie obrazka do projektu (189)

Kompletny kod układu (192)

Układy FrameLayout: podsumowanie (193)

Zabawy z widokami (201)

Pola tekstowe (202)

Przycisk (203)

Przycisk przełącznika (204)

Przełącznik (205)

Pola wyboru (206)

Przyciski opcji (208)

Lista rozwijana (210)

Widoki obrazów (211)

Dodawanie obrazów do przycisków (213)

Widoki przewijane (215)

Krótkie komunikaty (216)

Twój przybornik do Androida (220)

ROZDZIAŁ 6. Układy z ograniczeniami. Rozmieszczaj rzeczy w odpowiednich miejscach

Zagnieżdżone układy mogą być nieefektywne (222)

Przedstawiamy układy z ograniczeniami (223)

Nie zapomnij dołączyć do projektu biblioteki Constrained Layout Library (224)

Dodanie zasobów do strings.xml (225)

Zastosowanie narzędzia do tworzenia szkicu (226)

Rozmieszczanie widoków przy wykorzystaniu ograniczeń (227)

Dodawanie ograniczenia w pionie (228)

Zmiany szkicu są uwzględniane w kodzie XML (229)

Jak wyśrodkowywać widoki (230)

Zmiana położenia widoku poprzez określanie przesunięcia (231)

Jak zmieniać wielkość widoku? (232)

Jak wyrównywać widoki? (238)

Stwórzmy prawdziwy układ (239)

Zacznij od dodania widoków do górnego wiersza (240)

Mechanizm wnioskowania odgaduje, jakie ograniczenia należy dodać (241)

Dodaj do szkicu kolejny wiersz... (242)

I w końcu dodaj widok na treść wiadomości (243)

Jazda próbna aplikacji (244)

Twój przybornik do Androida (245)

ROZDZIAŁ 7. Widoki list i adaptery. Zorganizuj się

Każda aplikacja zaczyna się od pomysłu (248)

Użyj widoku listy do nawigowania po danych (251)

Aktywność szczegółów napoju (253)

Struktura aplikacji dla kafeтерии Coffeina (254)

Klasa Drink (256)

Układ aktywności głównego poziomu składa się z obrazka i listy (258)

Kompletny kod układu aktywności głównego poziomu (260)

Zapewnianie reakcji ListView na kliknięcia za pomocą obiektu nasłuchującego (261)

Dodanie obiektu nasłuchującego do widoku listy (262)

Aktywność kategorii wyświetla dane jednej kategorii (267)

Aktualizacja układu activity_drink_category.xml (268)

W przypadku danych statycznych należy użyć adaptera (269)

Łączenie widoków ListView z tablicami przy użyciu adaptera (270)

Dodanie adaptera ArrayAdapter do aktywności DrinkCategoryActivity (271)

Przegląd aplikacji, czyli dokąd dotarliśmy (274)

Jak obsługiwaliśmy kliknięcia w aktywności TopLevelActivity (276)

Kompletny kod aktywności DrinkCategoryActivity (278)

Wypełnienie widoków danymi (281)

Kod aktywności DrinkActivity (283)

Co się stanie po uruchomieniu aplikacji (284)

Twój przybornik do Androida (288)

ROZDZIAŁ 8. Biblioteki wsparcia i paski aplikacji. Na skróty

Świetne aplikacje mają przejrzystą strukturę (290)

Różne typy nawigacji (291)

Zacznijmy od paska akcji (293)

Utwórz aplikację Włoskie Co Nieco (295)

Dodaj bibliotekę wsparcia AppCompatActivity v7 (296)

Plik AndroidManifest.xml może zmieniać postać paska aplikacji (299)

Jak zastosować motyw? (300)

Zdefiniuj styl w pliku zasobów (301)

Dostosuj wygląd aplikacji (303)

Zdefiniuj kolory w pliku zasobów kolorów (304)

Kod pliku activity_main.xml (305)

Pasek aplikacji a pasek narzędzi (306)

Dołącz pasek narzędzi do układu aktywności (312)

Dodawanie akcji do paska aplikacji (315)

Zmień pasek aplikacji, dodając do niego etykietę (318)

Kod pliku AndroidManifest.xml (319)

Określ wygląd akcji (322)

Kompletny kod pliku MainActivity.java (325)

Włączanie nawigacji w górę (327)

Dzielenie się treściami z poziomu paska aplikacji (331)

Dodawanie dostawcy akcji udostępniania do menu_main.xml (332)

Określanie treści za pomocą intencji (333)

Kompletny kod aktywności MainActivity (334)

Twój przybornik do Androida (337)

ROZDZIAŁ 9. Fragmenty. Zadbaj o modularyzację

Twoja aplikacja musi wyglądać świetnie na WSZYSTKICH urządzeniach (340)

Może się zdarzyć, że aplikacja będzie musiała także działać inaczej (341)

Fragmenty umożliwiają wielokrotne stosowanie kodu (342)

Aplikacja w wersji na telefony (343)

Utworzenie projektu i aktywności (345)

Dodanie przycisku do układu aktywności (346)

Jak dodać fragment do projektu? (348)

Metoda onCreateView() fragmentu (350)

Dodawanie fragmentu do układu aktywności (352)

Zapewnienie interakcji fragmentu i aktywności (359)

Klasa Workout (360)

Przekazywanie identyfikatora treningu do fragmentu (361)

Określenie identyfikatora treningu w kodzie aktywności (363)

Cykl życia fragmentów (365)

Określenie zawartości widoków w metodzie onStart() fragmentu (367)

Jak utworzyć fragment typu ListFragment? (374)

Zaktualizowany kod klasy WorkoutListFragment (377)

Kod układu activity_main.xml (381)

Powiązanie listy z widokiem szczegółów (384)

Kod pliku WorkoutListFragment.java (387)

Aktywność MainActivity musi implementować interfejs (388)

Aktywność DetailActivity musi przekazać identyfikator do fragmentu WorkoutDetailFragment (389)

Twój przybornik do Androida (392)

ROZDZIAŁ 10. Fragmenty dla większych interfejsów. Różne wielkości, różne interfejsy

Nasza aplikacja Trenażer wygląda tak samo na telefonie i tablecie (394)

Projektowanie z myślą o większych interfejsach (395)

Wersja aplikacji na telefony (396)

Wersja aplikacji na tablety (397)

Utwórz AVD tabletu (399)

Umieszczaj zasoby przeznaczone dla różnych rodzajów ekranów w odpowiednich katalogach (402)

Różne opcje katalogów (403)

Tablety używają układów zapisanych w katalogu layout-large (408)

Jak działa zaktualizowany kod? (410)

Musimy zmienić kod metody itemClicked() (412)

Chcemy, by fragmenty współpracowały z przyciskiem Wstecz (413)

Witamy stos cofnięć (414)

Transakcje na stosie cofnięć nie muszą być aktywnościami (415)

Użyj układu FrameLayout, by programowo zmieniać fragmenty (416)

Skorzystaj z różnic w układach, aby określić, który z nich został użyty (417)

Zmodyfikowany kod aktywności MainActivity (418)

Stosowanie transakcji fragmentów (419)

Zaktualizowany kod aktywności MainActivity (423)

Zmiana orientacji tabletu wywołuje problem w aplikacji (427)

Zapisywanie stanu aktywności (po raz wtóry) (428)

Zaktualizowany kod pliku WorkoutDetailFragment.java (430)

Twój przybornik do Androida (432)

ROZDZIAŁ 11. Fragmenty dynamiczne. Zagnieżdżanie fragmentów

Dodawanie fragmentów dynamicznych (434)

Nowa wersja aplikacji (436)

Utwórz aktywność TempActivity (437)

Klasa TempActivity musi dziedziczyć po AppCompatActivity (438)

Kod fragmentu StopwatchFragment (444)

Układ fragmentu StopwatchFragment (447)

Dodanie fragmentu StopwatchFragment do układu aktywności TempActivity (449)

Atrybut onClick wywołuje metody aktywności, a nie fragmentu (452)

Powiązanie obiektu nasłuchującego OnClickListener z przyciskami (457)

Kod fragmentu StopwatchFragment (458)

Obrócenie urządzenia zeruje stoper (462)

Używaj dla statycznych fragmentów... (463)

W układzie activity_temp.xml zastosuj układ FrameLayout (464)

Kompletny kod aktywności TempActivity.java (467)

Dodanie stopera do fragmentu WorkoutDetailFragment (469)

Kompletny kod pliku WorkoutDetailFragment.java (476)

Twój przybornik do Androida (480)

ROZDZIAŁ 12. Biblioteka wsparcia wzornictwa. Przeciągnięcie w prawo

Aplikacja Włoskie Co Nieco w nowej odsłonie (482)

Struktura aplikacji (483)

Użycie klasy ViewPager do przewijania fragmentów (489)

Dodajemy ViewPager do układu aktywności MainActivity (490)

Przekaż kontrolce informacje o stronach przy użyciu odpowiedniego adaptera (491)

Kod naszego adaptera FragmentPagerAdapter (492)

Pełny kod pliku MainActivity.java (494)

Dodanie kart do aktywności MainActivity (498)

Jak dodać karty do układu? (499)

Połączenie układu kart z kontrolką ViewPager (501)

Pełny kod pliku MainActivity.java (502)

Biblioteka wsparcia wzornictwa pomaga implementować Material Design (506)

Zapewnienie reagowania paska narzędzi na przewijanie (508)

Dodanie CoordinatorLayout do układu aktywności MainActivity (509)

Jak koordynować przewijanie? (510)

Dodanie do fragmentu zawartości do przewijania (512)

Pełny kod pliku fragment_top.xml (515)

Dodanie zwijanego paska narzędzi do aktywności OrderActivity (517)

Jak stworzyć prosty zwijany pasek narzędzi? (518)

Jak dodać obrazek do zwijanego paska narzędzi? (523)

Aktualizacja kodu układu activity_order.xml (524)

Przyciski FAB i paski snackbar (526)

Zaktualizowany kod pliku activity_order.xml (528)

Pełny kod pliku OrderActivity.java (533)

Twój przybornik do Androida (535)

ROZDZIAŁ 13. Widoki RecyclerView i CardView. Stosuj recykling

Wciąż jest wiele do zrobienia w aplikacji Włoskie Co Nieco (538)

Widoki RecyclerView z wysokości 3000 metrów (539)

Dodanie danych pizz (541)

Wyświetlenie danych pizzy na karcie (542)

Jak utworzyć widok karty? (543)

Kompletny kod pliku card_captioned_image.xml (544)

Dodanie adaptera widoku RecyclerView (546)

Zdefiniowanie obiektu ViewHolder (548)

Przesłonięcie metody onCreateViewHolder() (549)

Dodanie danych do widoków CardView (550)

Kompletny kod pliku CaptionedImagesAdapter.java (551)

Utworzenie widoku RecyclerView (553)

Dodanie widoku RecyclerView do układu fragmentu PizzaFragment (554)

Kompletny kod pliku PizzaFragment.java (555)

RecyclerView rozmieszcza swoje widoki, używając menedżera układu (556)

Określanie menedżera układu (557)

Pełny kod fragmentu PizzaFragment.java (558)

Zapewnienie reakcji obiektu RecyclerView na kliknięcia (566)

Utworzenie aktywności PizzaDetailActivity (567)

Kod pliku PizzaDetailActivity.java (569)

Zapewnienie reakcji widoku RecyclerView na kliknięcia (570)

Można nasłuchiwać zdarzeń z widoków w adapterze (571)

Zapewnianie możliwości wielokrotnego stosowania adapterów (572)

Dodanie interfejsu do adaptera (573)

Implementacja interfejsu we fragmencie PizzaFragment (575)

Twój przybornik do Androida (578)

ROZDZIAŁ 14. Szuflady nawigacyjne. Z miejsca na miejsce

Widoki kart zapewniają łatwą nawigację... (580)

Planujemy utworzenie szuflady nawigacyjnej w nowej aplikacji pocztowej (581)

Szuflady nawigacyjne rozmontowane na czynniki pierwsze (582)

Utworzenie projektu Koci Czat (584)

Utworzenie fragmentu InboxFragment (585)

Utworzenie fragmentu DraftsFragment (586)

Utworzenie fragmentu SentItemsFragment (587)

Utworzenie fragmentu TrashFragment (588)

Przygotowanie układu paska narzędzi (589)

Aktualizacja motywu aplikacji (590)

Utworzenie aktywności HelpActivity (591)

Utworzenie aktywności FeedbackActivity (592)

Utworzenie nagłówka szuflady nawigacyjnej (594)

Kompletny kod pliku nav_header.xml (595)

Jak można grupować elementy? (598)

Sekcję wsparcia dodamy jako podmenu (600)

Kompletny kod pliku menu_nav.xml (601)

Jak utworzyć szufladę nawigacyjną? (602)

Kompletny kod układu aktywności activity_main.xml (603)

Dodanie fragmentu InboxFragment do układu aktywności MainActivity (604)

Dodanie przełącznika szuflady (607)

Reagowanie na klikanie elementów szuflady (608)

Implementacja metody onNavigationItemSelectedListener() (609)

Zamknięcie szuflady po naciśnięciu przycisku Wstecz (614)

Kompletny kod aktywności MainActivity (615)

Twój przybornik do Androida (619)

ROZDZIAŁ 15. Bazy danych SQLite. Odpal bazę danych

Znowu w kafeterii Coffeina (622)

Android trwale przechowuje dane, używając baz danych SQLite (623)

Android udostępnia kilka klas związanych z SQLite (624)

Obecna struktura aplikacji kafeterii Coffeina (625)

Zmienimy aplikację, by korzystała z bazy danych (626)

Pomocnik SQLite zarządza Twoją bazą danych (627)

Tworzenie pomocnika SQLite (628)

Wnętrze bazy danych SQLite (630)

Tabele stworzymy w języku SQL (631)

Wstawianie danych za pomocą metody insert() (632)

Wstawianie wielu rekordów (633)

Kod klasy CoffeinaDatabaseHelper (634)

Co robi kod pomocnika SQLite? (635)

Co zrobić, gdy trzeba będzie zmienić bazę? (636)

Bazy danych SQLite mają numer wersji (637)

Co się dzieje w przypadku zmiany numeru wersji? (638)

Aktualizacja bazy w metodzie onUpgrade() (640)

Przywracanie starszej wersji bazy za pomocą metody onDowngrade() (641)

Zaktualizujmy bazę danych (642)

Aktualizacja istniejącej bazy danych (645)

Aktualizacja rekordów za pomocą metody update() (646)

Stosowanie warunków odnoszących się do wielu kolumn (647)

Modyfikacja struktury bazy danych (649)

Usuwanie tabeli (650)

Pełny kod pomocnika SQLite (651)

Twój przybornik do Androida (656)

ROZDZIAŁ 16. Proste kursory. Pobieranie danych

Co się wydarzyło wcześniej... (658)

Struktura nowej wersji aplikacji kafeterii Coffeina (659)

Co zrobimy, by aktywność DrinkActivity zaczęła korzystać z bazy danych? (660)

Aktualny kod aktywności DrinkActivity (661)

Pobranie referencji do bazy danych (662)

Pobieranie danych z bazy za pomocą kursora (663)

Zwracanie wszystkich wierszy tabeli (664)

Zwracanie wierszy w określonej kolejności (665)

Zwracanie wybranych rekordów (666)

Dotychczasowy kod aktywności DrinkActivity (669)

Aby odczytać rekord z kursora, najpierw należy do niego przejść (670)

Poruszanie się po kursorze (671)

Pobieranie wartości z kursora (672)

Kod aktywności DrinkActivity (673)

Co udało się nam zrobić? (675)

Aktualny kod aktywności DrinkCategoryActivity (677)

Pobranie referencji do bazy danych kafeterii... (678)

Jak zastąpić tablicę przekazywaną do komponentu ListView? (679)

SimpleCursorAdapter odwzorowuje dane na widoki (680)

Stosowanie adaptera SimpleCursorAdapter (681)

Zamykanie kursora i bazy danych (682)

Ciąg dalszy opowieści (683)

Zmodyfikowany kod aktywności DrinkCategoryActivity (688)

Kod aktywności DrinkCategoryActivity (ciąg dalszy) (689)

Twój przybornik do Androida (691)

ROZDZIAŁ 17. Kursory i zadania asynchroniczne. Pozostając w tle

Chcemy, by nasza aplikacja aktualizowała dane w bazie (694)

Dodanie pola wyboru do układu aktywności DrinkActivity (696)

Wyświetlanie wartości kolumny FAVORITE (697)

Odpowiadanie na kliknięcia w celu aktualizacji bazy (698)

Kompletny kod aktywności DrinkActivity (701)

Wyświetlanie ulubionych napojów w aktywności TopLevelActivity (705)

Refaktoryzacja pliku TopLevelActivity.java (707)

Nowy kod aktywności TopLevelActivity (710)

Kursor można zmieniać za pomocą metody `changeCursor()` (715)

Który kod umieścić w którym wątku? (723)

Klasa `AsyncTask` służy do wykonywania operacji asynchronicznych (724)

Metoda `onPreExecute()` (725)

Metoda `doInBackground()` (726)

Metoda `onProgressUpdate()` (727)

Metoda `onPostExecute()` (728)

Parametry klasy `AsyncTask` (729)

Kompletny kod klasy `UpdateDrinkTask` (730)

Kompletny kod pliku `DrinkActivity.java` (732)

Twój przybornik do Androida (737)

Podsumowanie etapów działania zadań `AsyncTask` (737)

ROZDZIAŁ 18. Usługi uruchomione. Do usług

Usługi działają w tle (740)

Utworzymy usługę URUCHOMIONĄ (741)

Użycie klasy `IntentService` do utworzenia prostej usługi uruchomionej (742)

Jak rejestrować komunikaty? (743)

Kompletny kod usługi `DelayedMessageService` (744)

Usługi są deklarowane w pliku `AndroidManifest.xml` (745)

Dodajemy przycisk do układu `activity_main.xml` (746)

Usługę uruchamiamy, wywołując metodę `startService()` (747)

Stany usług uruchomionych (750)

Cykl życia usług uruchomionych: od utworzenia do usunięcia (751)

Nasza usługa dziedziczy metody cyklu życia (752)

Android dysponuje wbudowaną usługą obsługi powiadomień (755)

Użyjemy powiadomień z biblioteki wsparcia `AppCompat` (756)

W pierwszej kolejności tworzymy budowniczego powiadomień (757)

Wysyłanie powiadomień przy użyciu wbudowanej usługi systemowej (759)

Kompletny kod usługi DelayedMessageService (760)

Twój przybornik do Androida (765)

ROZDZIAŁ 19. Usługi powiązane i uprawnienia. Powiązane ze sobą

Usługi powiązane są skojarzone z innymi komponentami (768)

Utworzenie nowej usługi (770)

Zdefiniowanie obiektu Binder (771)

Dodanie metody getDistance() do usługi (772)

Aktualizacja układu aktywności MainActivity (773)

Utworzenie obiektu ServiceConnection (775)

Użycie metody bindService() do powiązania usługi (778)

Użycie metody unbindService() do odłączenia aktywności od usługi (779)

Wyświetlenie przebytego dystansu (780)

Kompletny kod aktywności MainActivity (781)

Stany usług powiązanych (787)

Dodanie biblioteki wsparcia AppCompat (790)

Dodanie do usługi OdometerService obiektu nasłuchującego danych o lokalizacji (792)

Zaktualizowany kod usługi OdometerService (795)

Wyliczenie przebytego dystansu (796)

Kompletny kod pliku OdometerService.java (798)

Jak poprosić o uprawnienia z poziomu aplikacji? (802)

Sprawdzenie odpowiedzi na prośbę (805)

Dodanie kodu wyświetlającego powiadomienia do metody onRequestPermissionsResult() (809)

Kompletny kod pliku MainActivity.java (811)

Twój przybornik do Androida (815)

Świetnie, że odwiedziliście nas w Androidowie (816)

DODATEK A. Układy względne i układy siatki. Poznaj krewnych

DODATEK B. Gradle. Program do budowy Gradle

DODATEK C. ART. Środowisko uruchomieniowe Androida

DODATEK D. ADB. Android Debug Bridge

DODATEK E. Emulator Androida. Przyspieszanie emulatora

DODATEK F. Pozostałości. Dziesięć najważniejszych zagadnień (których nie opisaliśmy)

1. Rozpowszechnianie aplikacji (862)
2. Dostawcy treści (863)
3. Klasy Loader (864)
4. Adaptery synchronizujące (864)
5. Odbiorcy komunikatów (865)
6. Klasa WebView (866)
7. Ustawienia (867)
8. Animacje (868)
9. Widżety aplikacji (869)
10. Testy zautomatyzowane (870)

Skorowidz (872)