

# РЕКОМЕНДУЄМО РЕСУРСИ ДЛЯ ВИВЧЕННЯ НАПРЯМКУ IOS

Одобрено NIX iOS experts

Починати створювати мобільні програми під iOS можна з вивчення популярну мову програмування Swift. Розробники Apple працювали кілька років, щоб створити просту, безпечну та зручну мову. Swift увібрав у себе найкращі ідеї сучасних мов із мудрістю інженерної культури Apple. Компілятор оптимізовано для продуктивності, а мова оптимізована для розробки, без компромісів з одного чи іншого боку. Swift доброзичливий по відношенню до новачків у програмуванні. Це перша мова програмування промислової якості, яка також зрозумілий і захоплюючий, як скриптова мова. Написання коду у пісочниці дозволяє експериментувати з кодом Swift і бачити результат миттєво, без необхідності компілювати та запускати програму. Ну що ж, уперед знайомитись зі світом яблучних технологій!

## ЗАГАЛЬНЕ

1. Знайомство з мовою варто розпочати з уважного вивчення офіційної [документації від Apple](#)

Особливу увагу зверніть на:

- керування та роботу з пам'яттю;
- класи та структури (особливості створення та використання);
- опціонали та роботу з ними, що це і навіщо;
- типи колекцій, у чому різниця та особливості використання;
- функції та замикання.

Інші особливості Swift не менш важливі для повноцінного програмування під iOS та їх розуміння буде плюсом для проходження навчання.

2. Обов'язково закріплюй отримані знання практикою та експериментами у [Swift Playgrounds](#)

## BEST PRACTICES AND BASICS

1. Розробникам-початківцям важливо добре розібратися з усіма
2. Основними засадами [ООП](#)
3. Буде великим плюсом зрозуміти якомога більше [патернів проектування](#).
4. Знання [SOLID](#) також буде плюсом.
5. Кожен програміст у своїй роботі стикається з багатопоточністю та конкурентним виконанням. Тому показником технічної грамотності буде знання основних концепцій з цієї галузі, які викладено у статті ["Multithreading and Concurrency Fundamentals"](#)

Не обмежуйся наведеними посиланнями: шукай різні джерела, щоб розібрати підходи з різних боків та на різних прикладах. Гугли чи вигадуй собі завдання у будь-якій предметній області, намагайся вирішити їх з використанням вивчених принципів.

## UI-КОМПОНЕНТИ

Важливо розібрати роботу якогось UI-ного компонента, наприклад UITableView, про них є безліч уроків, наприклад, як цей: ["Adding a UITableView programmatically"](#)

При цьому важливо розуміти:

- як малювати осередки у таблиці;
- як вони перевикористовуються;
- які обов'язкові методи необхідно реалізувати;
- у чому зручність та користь такого механізму;
- життєвий цикл програми, з якого стану переходить;
- життєвий цикл UIViewController;
- різниця UIWindow від UIView.

Корисно буде пробувати все на своєму маленькому тестовому додатку, щоб потім мати можливість показати код та розповісти чому застосовувалися ті чи інші рішення.

## ХОЧЕШ ДІЗНАТИСЯ БІЛЬШЕ?

- Підписуйся на наш [Telegram-канал](#) і завжди залишайся у курс нових вакансій для iOS Developer.
- Також запрошуємо тебе пройти безкоштовну [програму навчання](#) за напрямом iOS в NIX Solutions. Набір у групу відкрито!



## БУДЕМО РАДІ ЗУСТРІЧІ!

61002, Харків, вулиця Каразіна, 2  
jobs@nixsolutions.com  
+38 (067) 579-06-02, +38 (067) 575-23-26