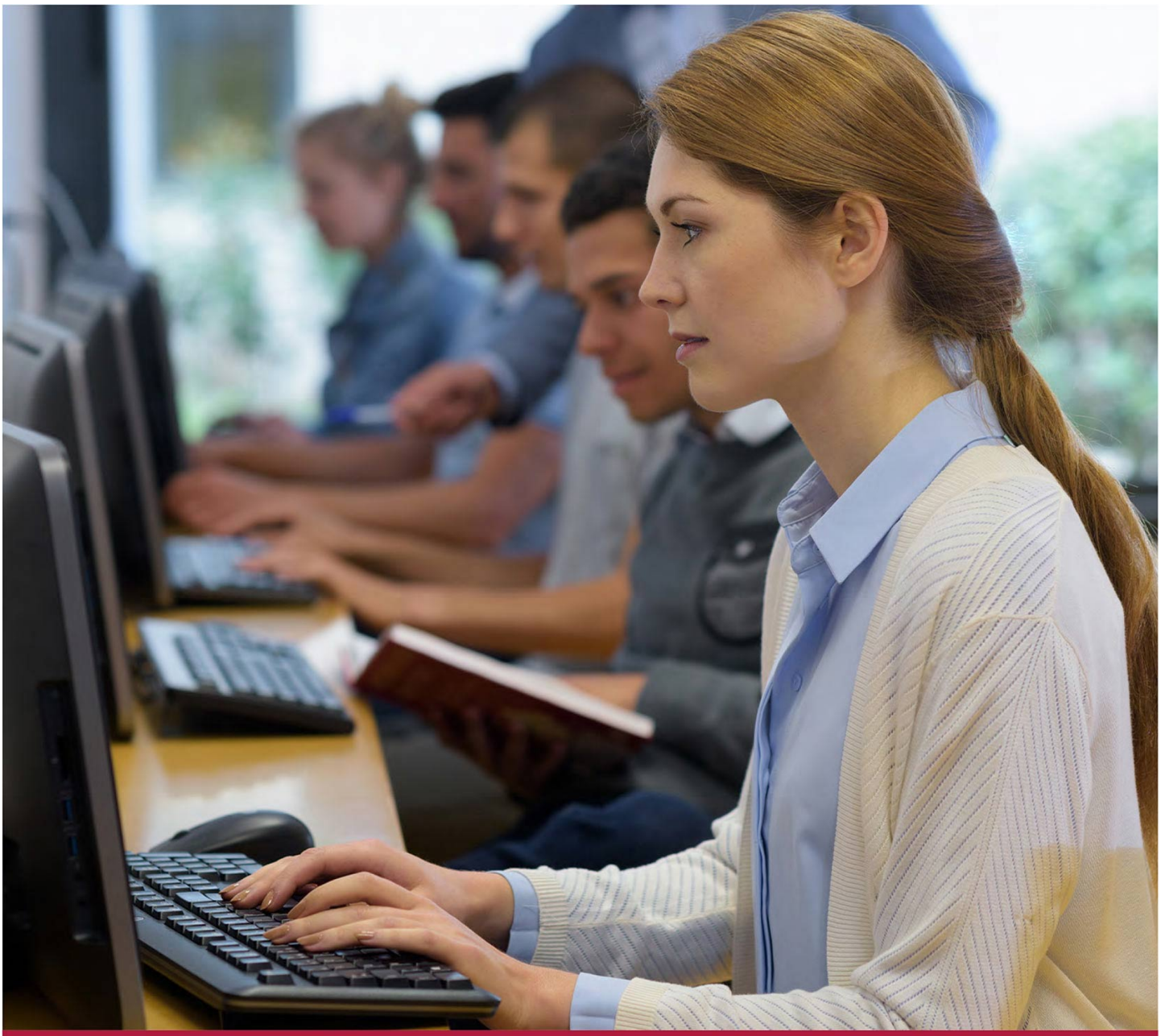
Академия АйТи
a Softline Company



Специалист DevSecOps

Код курса: pp_devsecops

+7 (495) 150 96 00 | academy@academyit.ru | academyit.ru

Специалист DevSecOps

Код курса: pp_devsecops

Длительность	276 ак. часов
Формат	
Разработчик курса	Академия АйТи
Тип	Учебный курс
Способ обучения	Под руководством тренера

О курсе

Актуальность программы вызвана необходимостью подготовки новых профессиональных кадров для Цифровой экономики Российской Федерации, так называемых DevSecOps инженеров, способных к внедрению безопасного инженерного процесса разработки современных цифровых платформ и экосистем на основе лучших международных практик разработки безопасного программного обеспечения (ПО). Кратко основные достоинства такой подготовки и переподготовки можно выразить в трех основных тезисах: скорость, безопасность и эффективность программного кода. Курс дополнен модулем по работе с ИИ в рамках безопасной разработки.

Подробная информация

Программа профессиональной переподготовки разработана на основании:

- Федерального законаот 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации»
- Приказа Министерства образования и науки Российской Федерации от 1 июля 2013 г. № 499 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам»
- Приказа Министерством труда и социальной защиты Российской Федерации от 12 апреля 2013 г. № 148н «О утверждении уровней квалификаций в целях разработки проектов профессиональных стандартов»
- Постановление Правительства Российской Федерации от 11 октября 2023 года N 1678 «Об утверждении Правил применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»
- ФГОС СПО 09.02.03 «Программирование в компьютерных системах», утвержденного приказом Министерства образования и науки РФ от 28.07.2014 г. №804
- Профессионального стандарта № 424н «Программист», утвержденный 20.07.2022 г.

Успешное окончание обучения по программе данного курса позволит специалистам:

- Взаимодействовать с командами разработки ПО и операционными подразделениями для анализа функциональных и не функциональных требований, пользовательских сценариев на

предмет безопасности;
- Заниматься подготовкой требований по безопасности, а также контролировать соблюдение требований в процессе разработки ПО;
- Проводить ревью безопасности разработанной архитектуры приложений (монолитной, SOA, микросервисной), ревью исходного кода ПО, тестирование безопасности релизов и систем в продакшене;
- Управлять требованиями по безопасности, в частности моделировать и анализировать угрозы безопасности;
- Управлять уязвимостью программных систем. Вести их учет, оценивать критичность и контролировать исправления в разработке ПО;
- Разрабатывать шаблоны безопасной архитектуры приложений, внедрять стандартны безопасного программирования, доводить требования безопасности до команд разработки ПО;
- Внедрять лучшие практики разработки безопасного ПО, DevSecOps (SAST, DAST, dependency check и др);
- Своевременно информировать команды разработчиков ПО об обнаружении новых и ранее неизвестных уязвимостях программного кода и предлагать обоснованные решения для их нейтрализации.

Успешное окончание обучения по программе данного курса позволит специалистам:

Знать:

- основные методы и практики DevSecOps;
- методы и приемы алгоритмизации поставленных задач безопасной разработки ПО;
- основные структуры данных Python;
- алгоритмы решения типовых задач, области и способы их применения;
- синтаксис выбранного языка Python;
- особенности программирования на Python;
- стандартные библиотеки языка Python;
- технологии безопасного программирования;
- особенности выбранной среды программирования;
- методы и приемы отладки программного кода;
- типы и форматы сообщений об ошибках и уязвимостях ПО;
- методологии разработки компьютерного программного обеспечения;
- основные стандарты оформления технической документации на компьютерное программное обеспечение;

Уметь:

- применять Python для написания безопасного программного кода;
- использовать выбранную среду программирования;
- отлаживать программный код;
- использовать вспомогательные инструментальные программные средства для обработки исходного текста программного кода;
- выявлять ошибки в программном коде;
- контролировать компоненты с открытым исходным кодом при попадании в периметр разработки (Open Source Analysis, OSA);
- проводить статический анализ кода (Static Application Security Testing, SAST);
- контролировать состав компонент ПО (Software Composition Analysis, SCA);

- проводить динамический анализ кода (Dynamic Application Security Testing, DAST/Interactive Application Security Testing, IAST/Behavioral Application Security Testing, BAST);
- проводить анализ бинарного кода и контроль состава контейнеров (Bytecode and Container Analysis, BCA);
- применять известные технические меры безопасности для разработки безопасного ПО;
- применять алгоритмы решения типовых задач в соответствующих областях;
- использовать возможности имеющейся технической и/или программной архитектуры для написания программного кода;
- применять заданные стандарты и шаблоны для составления и оформления технической документации

Владеть навыками / выполнять следующие трудовые действия:

- составление формализованных описаний решений поставленных задач по разработке безопасного ПО в соответствии с требованиями технического задания или других принятых в организации нормативных документов;
- создание безопасного программного кода в соответствии с техническим заданием (готовыми спецификациями);
- оптимизация программного кода с использованием специализированных программных средств DevSecOps;
- приведение наименований переменных, функций, классов, структур данных и файлов в соответствие с установленными в организации требованиями по безопасности;
- структурирование исходного программного кода в соответствии с установленными в организации требованиями по безопасности;
- форматирование исходного программного кода в соответствии с установленными в организации требованиями по безопасности;
- слияние, разделение и сравнение исходных текстов безопасного программного кода;
- анализ и проверка исходного программного кода на требования по безопасности;
- разработки алгоритмов решения поставленных задач в соответствии с требованиями технического задания или внутренних документов организации;
- оформление технической документации на компьютерное программное обеспечение по заданному стандарту или шаблону.

Получение компетенций для профессиональной деятельности в области программирования и создания ИТ- продуктов:

- Формализация и алгоритмизация поставленных задач для разработки программного кода
- Написание программного кода с использованием языков программирования, определения и манипулирования данными в базах данных
- Оформление программного кода в соответствии с установленными требованиями
- Работа с системой управления версиями программного кода
- Проверка и отладка программного кода
- Разработка тестовых наборов данных для проверки работоспособности компьютерного программного обеспечения
- Проверка работоспособности компьютерного программного обеспечения
- Разработка процедур интеграции программных модулей
- Осуществление интеграции программных модулей и компонентов и проверки работоспособности выпусков программного продукта
- Анализ возможностей реализации требований к компьютерному программному обеспечению

- Разработка технических спецификаций на программные компоненты и их взаимодействие
- Проектирование компьютерного программного обеспечения.

Целевая аудитория:

- DevOps-инженеры.
- Архитеторы ПО
- Программисты
- Специалисты по ИБ.

Необходимая подготовка: Знание основ программирования на языке Python.

Программа курса

Модуль 1 «Основы программирования и составления алгоритмов»

- Алгоритм – свойства и способы представления. Типы данных – назначение и роль в программе. Операнды и операторы – вычисление выражений.
- Модели разработки программ. Структурное программирование. Базовые принципы: блочная структура кода – блоки и подпрограммы.
- Практические примеры составления блок-схем и псевдокода. Простейшие алгоритмические задачи. Перевод алгоритма в код. Подпрограммы (функции) как основные блоки кода. Типовые задачи на обработку текста.
- Установка интерпретатора Python. Основные операции и типы данных. Особенности ввода и вывода. Операторы ветвления и циклы. Установка и запуск среды разработки. Типы данных: числа, строки, списки, логический тип, None. Функции преобразования типов. Простой ввод и простой вывод.
- Ветвления. Оператор if. Базовая форма цикла while. Операторы break и continue. Перебор (for).
- Строки. Обработка строк.
- Методы и функции. Виды переменных. Вложенные функции. Лямбда выражения. Список. Основные операции со списком. Кортеж. Основные операции с кортежем. Распаковка кортежа. Словарь. Основные операции со словарем. Множества. Основные операции с множеством. Работа с массивами.
- Модули и пакеты. Модуль Decimal, OS. Работа с файловой системой.
- Регулярные выражения.
- Взаимодействие с CSV, XML, JSON.
- Обработка исключений. Обработка исключений. Создание исключений. Стандартные исключения.
- Взаимодействие с реляционными базами данных. Основы SQL, сложные запросы, проектирование баз данных, нормализация. Транзакции. Уровни изоляции транзакции. Требования ACID.
- Основы проектирования приложений. Основы ООП.
- Основы работы с GIT.

Модуль 2 «Разработка безопасного ПО»

- Тема 1. Основные концепции DevSecOps.
- Тренды разработки современного ПО (внедрение практик DevOps, снижение общих сроков

разработки ПО (time to market), повышение гибкости в разработке ПО, переход от монолитных к микросервисным приложениям, динамическое выделение ИТ-ресурсов, повышение внимания к вопросам разработки безопасного ПО).
- Основные понятия и определения DevOps и DevSecOps.
- Ключевые компоненты DevSecOps (анализ кода, управление изменениями, мониторинг соотвествия, иследование угроз безопасности, оценка уязвимости кода, обучение и повышение осведомленности).
- Основные этапы DevSecOps (разработка приложения и работа с репозиторием программ, непрерывная интеграция (CI) и тестирование приложения, непрерывное развертывание (CD) приложения в рабочей среде, контроль новой версии приложения в рабочей среде).
- Возможные сценарии интеграции DevSecOps в процессы и инфраструктуру компании.
- Адаптации функции кибербезопасности и интеграции DevSecOps.
- Проведение оценки безопасности процесса разработки, а также идентификация ключевых рисков и риск-факторов, связанных с недостатками процесса.
- Ключевые точки процесса разработки, где необходимо включение мер безопасности
- Информирование сотрудников о критических рисках и мерах для их снижения.
- Тема 2. Основные практики DevSecOps.
- Контроль компонент с открытым исходным кодом (Open Source Analysis, OSA)
- Статический анализ кода (Static Application Security Testing, SAST).
- Контроль состава компонент ПО (Software Composition Analysis, SCA).
- Динамический анализ кода (Dynamic Application Security Testing, DAST/Interactive Application Security Testing, IAST/Behavioral Application Security Testing, BAST).
- Фаззинг - как метод исследования уязвимостей.
- Фаззинг-тестирование (fuzzing).
- Практика по использованию сканеров уязвимостей (выполняется под непосредственным руководством преподавателя). Решения для выявления и нейтрализации уязвимостей программного кода.
- Тема 3. Трансформация DevOps в DevSecOp.
- Трансформация DevOps в DevSecOps. Использование безопасных по умолчанию библиотек, фреймворков и компонент ПО в процессе разработки (Secure-by-Default).
- Интеграция технологических практик ИБ в начало конвейера CI/CD (Shift-Left подход).
- Автоматизация процессов в концепции Everything-as-a-Code.
- Формирование сообщества security-чемпионов в производственных командах для повышения инженерной security-культуры.
- Применение модели зрелости DevSecOps для оценки существующего процесса и для постоянного совершенствования.
- Обеспечение прозрачности security активностей для участников инженерного производственного процесса.
- DevSecOps-оркестрация (Application Security Testing Orchestration, ASTO) для непрерывного улучшения процесса разработки безопасного ПО.
- Тема 4. Национальные требования (ГОСТ Р 56939-2016 и ГОСТ Р ИСО/МЭК 12207) в части разработки безопасного ПО.
- Требования в части идентификации и аутентификации.
- Требования по защите от несанкционированного доступа к информации.
- Требования в части регистрации событий безопасности.
- Требования контроля точности, полноты и правильности входных и выходных данных.
- Требования по обработке программных ошибок и исключительных ситуаций.
- Требования класса ASE ""Оценка задания по безопасности"" (ГОСТ Р ИСО/МЭК 15408-3).
- Меры по разработке безопасного программного обеспечения, реализуемые при выполнении

инсталляции программы и поддержки приемки программного обеспечения.
- Меры по разработке безопасного программного обеспечения, реализуемые при решении проблем в программном обеспечении в процессе эксплуатации.
- Меры по разработке безопасного программного обеспечения, реализуемые в процессе менеджмента инфраструктурой среды разработки программного обеспечения.
- Тема 5. Роли и кадровое обеспечение DevSecOps.
- Принятие методологии разработки безопасного ПО, DevSecOps (оценка текущих мер безопасности и распределение ролей в команде разработки ПО, внедрение мер безопасности на стадии проектирования программных систем, внедрение инструментов и авто-тестов безопасности в пайплайн, тестирование безопасности разработанных решений, анализ выявленных уязвимостей и подготовка рекомендаций по их устранению, обучение лучшим практикам разработки безопасного ПО, Best Practices).
- Распределение ролей в процессе DevSecOps (продакт менеджер, архитектор, команда разработки, QA, AppSec специалист, DevOps инженер).
- Воспитание чемпионов безопасности Security Champions (масштабирование безопасности с помощью нескольких команд разработки, привлечение сотрудников, не связанных с безопасностью, но связанных с DevOps, создание и развитие культуры безопасности).
- Повышение осведомленности по вопросам разработки безопасного ПО.
- Развитие корпоративной культуры DevSecOps и безопасности в целом.

Модуль 3. Инструменты искусственного интеллекта в работе специалиста DevSecOps

- Тема 1. Что представляет из себя ИИ на современном этапе развития
- Тема 2. Большие языковые модели, логика их работы и возможности
- Тема 3. Способы применения больших языковых моделей
- Тема 4. Специфика ИИ-агента в DevSecOps
- Тема 5. Классификация сценариев по этапам: Plan → Code → CI → Deploy → Operate
- Тема 6. Shift-left security с ИИ
- Тема 7. Управление рисками при внедрении ИИ-агентов
- Тема 8. Разбор применения на примере тестового кейса

Посмотреть расписание курса и записаться на обучение

Обращайтесь по любым вопросам
к менеджерам Академии АйТи

+7 (495) 150 96 00 | academy@academyit.ru