

# Beap и его жизненный цикл

Даниил Пилипенко

Директор центра подбора IT-специалистов  
SymbioWay

Skillbox

# Понятие бина (bean)

Объект или компонент, управляемый контейнером Spring. Контейнер создаёт, настраивает и управляет жизненным циклом бина.

Бин может быть любым классом, который несёт определённое значение или функциональность в приложении.

# Способы поиска Bean

- **Аннотации.** Вы можете использовать аннотации, такие как `@Component`, `@Service`, `@Repository` и другие из пакета `org.springframework.stereotype`, чтобы пометить классы как компоненты. Spring сканирует пакеты вашего приложения на наличие классов с этими аннотациями и регистрирует их как бины
- **Явное указание бинов.** Вы можете явно указать бины в конфигурационных классах с помощью аннотации `@Bean`. Это позволяет создавать бины на основе определённых методов
- **XML-конфигурация.** В случае использования XML-конфигурации вы можете определить бины в файлах конфигурации, указывая классы вручную

# Жизненный цикл бина

- Инициализация (Initialization)
- Внедрение зависимостей
- Выполнение методов инициализации
- Использование (Using)
- Уничтожение (Destruction)
- Освобождение ресурсов
- Удаление из контекста

# Область видимости Bean

**Singleton (Одиночка).** Бины с областью видимости singleton создаются только один раз в контексте приложения и существуют на протяжении всего его жизненного цикла. Они широко используются для создания общих ресурсов, кешей и других объектов, которые можно безопасно разделять между различными компонентами приложения.

**Prototype (Прототип).** Бины с областью видимости prototype создаются каждый раз при запросе. Это означает, что каждый новый запрос к бину будет возвращать новый экземпляр объекта. Прототип-бины полезны в случаях, когда требуется изолировать состояние между разными запросами или когда объекты имеют короткоживущий характер и должны создаваться заново для каждого запроса.

# Область видимости Bean

**Session (Сессия).** Бины с областью видимости `session` создаются один раз для каждой сессии пользователя в веб-приложении. Они позволяют хранить данные, специфичные для сеанса пользователя, и обеспечивают изоляцию между разными пользователями.

**Request (Запрос).** Бины с областью видимости `request` создаются для каждого HTTP-запроса в веб-приложении. Это позволяет хранить данные, специфичные для текущего запроса, и гарантирует изоляцию между разными запросами.