

Сборка и запуск проекта

Запуск Docker-контейнеров:

```
docker-compose build --no-cache && docker-compose up -d
```

Остановка контейнеров:

```
docker-compose down
```

Пересборка сервиса (например, PHP):

```
docker-compose build php && docker-compose up -d --no-deps php
```

Frontend (React)

Установка зависимостей:

```
npm install
```

Запуск в режиме разработки:

```
npm run dev
```

Продакшен-сборка:

```
npm run build
```

Анализ размера бандла:

```
npm run analyze
```

Backend (Laravel)

Установка PHP-зависимостей (в контейнере):

```
docker-compose exec php composer install
```

Генерация ключа приложения:

```
docker-compose exec php php artisan key:generate
```

Миграции + сидеры:

```
docker-compose exec php php artisan migrate --seed
```

Очистка кэша:

```
docker-compose exec php php artisan optimize:clear
```

Тестирование

Jest-тесты (React):

```
npm test
```

Тесты с покрытием кода:

`npm test -- --coverage`

PHPUnit-тесты (Laravel):

`docker-compose exec php php artisan test`

Запуск конкретного теста:

`docker-compose exec php php artisan tests/Feature/AuthTest.php`

Управление Docker

Просмотр логов контейнера:

`docker logs -f invest-calc-php`

Список запущенных контейнеров:

`docker ps`

Очистка неиспользуемых данных Docker:

`docker system prune -f`

Полная пересборка (с удалением volume'ов):

`docker-compose down -v && docker-compose up -d`

Production-развертывание

Сборка production-образов:

`docker-compose -f docker-compose.yml -f docker-compose.prod.yml build`

Запуск в production:

`docker-compose -f docker-compose.yml -f docker-compose.prod.yml up -d`

Миграции на production:

`docker-compose exec php php artisan migrate --force`

Полезные комбинации

Полная пересборка фронта и бэка:

`npm run build && docker-compose exec php php artisan optimize`

Рестарт проекта:

`docker-compose down && docker-compose up -d --build`

Запуск всех тестов + линтер:

```
npm test && docker-compose exec php php artisan test && npm run lint
```

Аварийные ситуации

Сброс node_modules:

```
rm -rf node_modules && npm install
```

Исправление прав в Laravel:

```
docker-compose exec php chmod -R 775 storage bootstrap/cache
```

Переустановка composer-зависимостей:

```
docker-compose exec php rm -rf vendor && composer install
```

Дополнительные инструменты

Проверка стиля кода (ESLint):

```
npm run lint
```

Автоисправление кода:

```
npm run lint:fix
```

Проверка устаревших зависимостей:

```
npm outdated
```

```
docker-compose exec php composer outdated
```
