

# Тест[[1]](#footnote-1) для соискателей вакансии

**Junior Python Разработчик**

Уважаемый соискатель, перед Вами тестовое задание на вакансию «Junior Python Разработчик» в компании Saber Interactive.

Индустрия видеоигр, в которой работает компания Saber, предъявляет высокие требования к качеству, эффективности и удобству восприятия программного кода. Поэтому мы предлагаем выполнить Вам задание, чтобы Вы могли продемонстрировать Вашу способность писать такой код.

**Билд-система**

***Описание***

Для разработки игр требуется билд-система, которая автоматизирует и ускоряет рутинные процессы.

Система оперирует понятиями “задача” и “билд”.

**Задача** – это то, что нужно сделать.

Например:

* собрать ресурсы игры;
* скомпилировать .exe;
* запаковать игру;

и так далее.

Задача описывается уникальным именем (`name`) и ее зависимостями (`dependencies`) от других задач. Задача не может быть выполнена раньше, чем ее зависимости.

Описания задач задаются в yaml-файле `[tasks.yaml](#_Приложение_1._tasks.yaml)` (в приложении к заданию).

**Билд** – это группа задач, объединенных функционально.

Например:

* "собрать игру" с задачами: "собрать ресурсы игры", "скомпилировать .exe", "запаковать игру";
* "запустить тесты" с задачами: "собрать ресурсы игры", "скомпилировать .exe";

и так далее.

Билд описывается уникальным именем (`name`) и списком задач (`tasks`).

Описания билдов задаются в yaml-файле `[builds.yaml](#_Приложение_2._builds.yaml)` (в приложении к заданию).

***Требования***

1. Используя Python, написать CLI-утилиту для определения последовательности выполнения задач в билде. Допускается применять готовые пакеты для обработки аргументов. Например — [docopt](http://docopt.org/).
2. Программа должна загрузить tasks.yaml и builds.yaml, которые содержат **задачи** и **билды** соответственно, и дальше оперировать ими. По умолчанию, путь до файлов считать текущей рабочей директорией, с возможностью переопределения опционально.
3. Программа должна поддерживать:

* Команду **list** с аргументами **builds** и **tasks** для просмотра имен загруженных билдов и задач соответственно.

Примеры:

“””

app.py list builds

List of available builds:

 \* test\_game

 \* pack\_game

“””

“””

app.py list tasks

List of available tasks:

 \* compile\_exe

 \* compile\_shaders

 \* pack\_build

“””

* Команду **get** с аргументами **build <build\_name>** и **task <task\_name>** для получения детальной информации.

Примеры:

“””

app.py get task pack\_build

Task info:

 \* name: pack\_build

 \* dependencies: compile\_exe, compile\_shaders

“””

“””

app.py get build pack\_game

Build info:

 \* name: pack\_game

 \* tasks: compile\_exe, compile\_shaders, pack\_build

“””

При выводе задач билда должны учитываться зависимости задач (т.е. сначала должны выводится зависимости, а затем — зависящие от них задачи).

***Практические результаты***

1. Исходный код программы.
2. Файл README.md с инструкциями по запуску и тестированию программы.

***Критерии оценки***

1. Качество кода, удобочитаемость и организация.
2. Правильная обработка ошибок и крайних случаев.
3. Наличие тестов, % покрытия кода.

## 

## ***Приложение 1. tasks.yaml***

tasks:

- name: bring\_black\_leprechauns

  dependencies: []

- name: bring\_gray\_cyclops

  dependencies: []

- name: bring\_green\_cyclops

  dependencies: []

- name: bring\_purple\_leprechauns

  dependencies: []

- name: bring\_yellow\_cyclops

  dependencies: []

- name: build\_blue\_leprechauns

  dependencies:

  - bring\_purple\_leprechauns

- name: build\_lime\_cyclops

  dependencies: []

- name: coloring\_green\_cyclops

  dependencies:

  - bring\_gray\_cyclops

  - enable\_white\_cyclops

  - read\_lime\_cyclops

- name: create\_green\_cyclops

  dependencies:

  - bring\_green\_cyclops

  - design\_silver\_cyclops

  - enable\_yellow\_cyclops

  - read\_aqua\_cyclops

  - train\_white\_cyclops

- name: create\_white\_cyclops

  dependencies: []

- name: design\_black\_centaurs

  dependencies: []

- name: design\_olive\_cyclops

  dependencies:

  - coloring\_green\_cyclops

  - create\_green\_cyclops

  - design\_teal\_cyclops

- name: design\_silver\_cyclops

  dependencies: []

- name: design\_teal\_cyclops

  dependencies:

  - bring\_yellow\_cyclops

  - build\_lime\_cyclops

  - create\_white\_cyclops

  - map\_black\_cyclops

- name: enable\_fuchsia\_fairies

  dependencies: []

- name: enable\_white\_cyclops

  dependencies: []

- name: enable\_yellow\_cyclops

  dependencies: []

- name: map\_black\_cyclops

  dependencies: []

- name: map\_gray\_centaurs

  dependencies:

  - read\_purple\_centaurs

  - train\_silver\_centaurs

- name: read\_aqua\_cyclops

  dependencies: []

- name: read\_blue\_witches

  dependencies: []

- name: read\_lime\_cyclops

  dependencies: []

- name: read\_purple\_centaurs

  dependencies: []

- name: train\_silver\_centaurs

  dependencies:

  - design\_black\_centaurs

  - upgrade\_blue\_centaurs

- name: train\_white\_cyclops

  dependencies: []

- name: upgrade\_blue\_centaurs

  dependencies: []

- name: upgrade\_lime\_leprechauns

  dependencies:

  - bring\_black\_leprechauns

  - build\_blue\_leprechauns

  - write\_lime\_leprechauns

- name: upgrade\_olive\_gnomes

  dependencies: []

- name: upgrade\_olive\_leprechauns

  dependencies: []

- name: write\_aqua\_leprechauns

  dependencies: []

- name: write\_lime\_leprechauns

  dependencies:

  - upgrade\_olive\_leprechauns

  - write\_aqua\_leprechauns

## ***Приложение 2. builds.yaml***

builds:

- name: approach\_important

  tasks:

  - map\_gray\_centaurs

- name: audience\_stand

  tasks:

  - enable\_fuchsia\_fairies

  - read\_blue\_witches

  - upgrade\_olive\_gnomes

- name: time\_alone

  tasks:

  - design\_olive\_cyclops

  - upgrade\_lime\_leprechauns

1. Данное задание разработано специально для тестирования кандидатов и не имеет отношения к какому-либо конкретному проекту, поэтому мы гарантируем, что ваш концепт арт, рисунок, код, модель и т.д., выполненный в качестве теста не будет использоваться в коммерческой разработке. [↑](#footnote-ref-1)