



## Zendesk Coding Challenge

### OVERVIEW

Using the provided data (`tickets.json` and `users.json` and `organization.json`) write a simple command line application (or a locally runnable web-app) to search the data and return the results in a human readable format.

When executing a search operation, where data exists, values from any related entities should be included in the results. The user should be able to search on any field and exact value matching is fine (e.g. “mar” won’t return “mary”). The user should also be able to search for empty values, e.g. where description is empty.

Search can get pretty complicated pretty easily, we just want to see that you can code a basic search application.

### EVALUATION CRITERIA

We will look at your project and assess it for:

1. **Does It Work** - Does the solution run per the instructions?
2. **Design** - Does the solution demonstrate good modularity, extensibility, and separation of concerns? Are appropriate OOP/OOD principles employed?
3. **Readability** - Does the solution display proficiency with the language, a consistent style, and is it understandable? **Hint:** Use of your favorite linter or static analysis tool highly encouraged!
4. **Testing** - Do the tests pass, do they verify all the use cases, are they structured well, do they break when unexpected changes are made?
5. **Robustness** - Does the solution work quickly, efficiently, and gracefully handle unexpected situations?

If you have any questions about this criteria please ask.

## SPECIFICATIONS

- Use the language in which you are strongest.
- Feel free to use libraries or roll your own code as you see fit.
- Include a README with (accurate) usage instructions.

## SUBMISSION

Github is the preferred option (a public repo is fine) but we will also accept a .zip file if necessary. Email your solution to [kraza@zendesk.com](mailto:kraza@zendesk.com).

## HINTS

- A member of our team wrote a blog about how to pass a coding challenge:  
<http://chocolatetin.org/2015/08/08/how-to-pass-a-coding-test.html>.
- The problem definition is pretty basic – if you wish to implement additional (or expanded) functionality that would be awesome – please make note of any assumptions or design decisions you make.

## SAMPLE OUTPUT

*Note: This output is not prescriptive, in fact we encourage you to do better!*

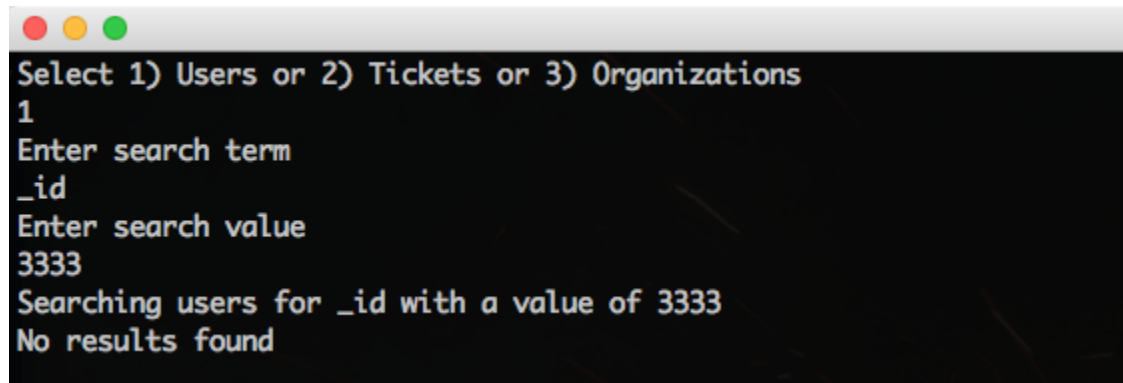
### CLI - Display Results

```
1. ruby
Welcome to Zendesk Search
Type 'quit' to exit at any time, Press 'Enter' to continue

Select search options:
* Press 1 to search Zendesk
* Press 2 to view a list of searchable fields
* Type 'quit' to exit

1
Select 1) Users or 2) Tickets or 3) Organizations
1
Enter search term
_id
Enter search value
71
_id 71
url http://initech.zendesk.com/api/v2/users/71.json
external_id c972bb41-94aa-4f20-bc93-e63dbfe8d9ca
name Prince Hinton
alias Miss Dana
created_at 2016-04-18T11:05:43 -10:00
active true
verified false
shared false
locale zh-CN
timezone Samoa
last_login_at 2013-05-01T01:18:48 -10:00
email danahinton@flotonic.com
phone 9064-433-892
signature Don't Worry Be Happy!
organization_id 121
tags ["Davenport", "Cherokee", "Summertown", "Clinton"]
suspended false
role agent
organization_name Hotcakes
ticket_0 A Catastrophe in Micronesia
ticket_1 A Drama in Wallis and Futuna Islands
ticket_2 A Drama in Australia
```

## CLI - No Search Result

A terminal window with a dark background and a light gray title bar containing three colored window control buttons (red, yellow, green). The text is displayed in a monospaced font. The sequence of text is: 'Select 1) Users or 2) Tickets or 3) Organizations', followed by '1' on the next line. Then 'Enter search term' followed by '\_id' on the next line. Then 'Enter search value' followed by '3333' on the next line. Then 'Searching users for \_id with a value of 3333' followed by 'No results found' on the final line.

```
Select 1) Users or 2) Tickets or 3) Organizations
1
Enter search term
_id
Enter search value
3333
Searching users for _id with a value of 3333
No results found
```

## CLI - List Searchable Fields

```
-----  
Search Users with  
_id  
url  
external_id  
name  
alias  
created_at  
active  
verified  
shared  
locale  
timezone  
last_login_at  
email  
phone  
signature  
organization_id  
tags  
suspended  
role  
  
-----  
Search Tickets with  
_id  
url  
external_id  
created_at  
type  
subject  
description  
priority  
status  
recipient  
submitter_id  
assignee_id  
organization_id  
tags  
has_incidents  
due_at  
via  
requester_id  
  
-----  
Search Organizations with  
_id  
url  
external_id  
name  
domain_names  
created_at  
details  
shared_tickets  
tags
```