Dear candidate,

Thank you for choosing SmartCat as a company where you would like to work and where we could grow together. People are the core of our company and we encourage everyone with software engineering skills to take a chance and enter our selection process.
You have approximately 16 hours to complete the challenge that is given to you. Please read the instructions carefully. You are allowed to use any online resources. After completing the challenge, please compress your work directory and send us via email or share it on a public git repository. Please write clear instructions on how to run the application.

Treat this test as confidential and do not share it.

Thank you and good luck!

Work shifts challenge

Description

Data generator

Create a REST endpoint that simulates data collected about work shift. The specs for endpoint are here: https://my.tanda.co/api/v2/documentation#shifts
The endpoint should generate random shift data for the previous week. The endpoint should just return the data, no need to back it with the database. No need to provide authentication.

Shift: 326872

```
[
 {
  "id": 326872,
  "timesheet_id": 47237,
  "user_id": 3274,
  "date": "2020-07-23",
  "start": 1595526660,
  "breaks": [
   {
    "id": 96575,
    "shift_id": 326872,
    "start": 1595544960,
    "finish": 1595546160,
    "length": 20,
    "paid": false,
    "updated_at": 1595546174
   }
  ],
  "finish": 1595563380,
  "department_id": 2628,
  "sub_cost_centre": null,
  "tag": null,
  "tag_id": null,
  "status": "PENDING",
  "metadata": null,
  "leave_request_id": null,
  "allowances": [
   {
    "id": 180,
    "name": "NY Spread of Hours (Other NY State $11.80/hr)",
    "value": 1.0,
      "updated_at": 1595563362,
      "cost": 11.8
    }
  ],
  "shift_feedback_id": null,
  "approved_by": null,
  "approved_at": null,
  "award_interpretation": [
    {
      "units": 1.0,
      "date": "2020-07-23",
      "export_name": "SOH",
      "secondary_export_name": null,
      "ordinary_hours": null,
      "cost": 11.8
    },
    {
      "units": 4.7334,
      "date": "2020-07-23",
      "export_name": "R",
      "secondary_export_name": null,
      "ordinary_hours": true,
      "cost": 65.32092,
      "from": 1595546160,
      "to": 1595476800
    },
    {
      "units": 5.0833,
      "date": "2020-07-23",
      "export_name": "R",
      "secondary_export_name": null,
      "ordinary_hours": true,
      "cost": 70.14954,
      "from": 1595526660,
      "to": 1595544960
    },
    {
      "units": 0.05,
      "date": "2020-07-24",
      "export_name": "R",
      "secondary_export_name": null,
      "ordinary_hours": true,
      "cost": 0.69,
      "from": 1595563200,
      "to": 1595563380
    }
  ],
  "cost": 147.96046,
  "cost_breakdown": {
   "award_cost": 136.16046,
   "allowance_cost": 11.8
  },
  "updated_at": 1595563361,
  "record_id": 792126,
  "last_costed_at": 1595563361
 }
]
```

ETL Job

Create an ETL job that fetches the data from the endpoint from step 1, transforms the data and loads it into a database of your choice (postgres, mongo, etc). The resulting database should contain 4 tables:

1. breaks - which containc all the `breaks` fetched from the shift data from the API.
2. allowances - which contains all the `allowances` fetched from the shift data from the API
3. award_interpretations - which contains all the `award_interpretations` fetched from the shift data from the API
4. shifts - which should contain everything it does except for breaks, allowances, awar_interpretation properties (arrays);
5. All the timestamps should be converted to EST timezone;
6. breaks, allowances, award_interpretation should be enriched with shift_id (corresponds to ‘id’ column in the shift object), shift_date (corresponds to ‘date’ in shift object), sheet_id (corresponds to ‘sheet_id’ in shift object);

Deliverables

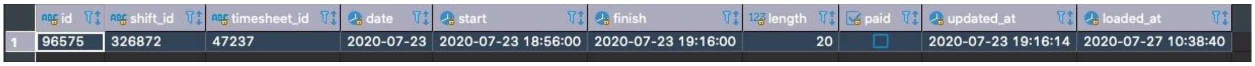
- Working or even non-working code sent in zip archive or shared via public git repository
- Code should be written in Java or Python
- Code should contain README file that explains the approach and how to run the applications
- Deployment and cleanup should be as simple as possible

General advice

- Use common sense
- Keep things simple
- It’s much better to have a working solution than the perfect, but not working solution
- Bonus points: cover the ETL with an automated test suite

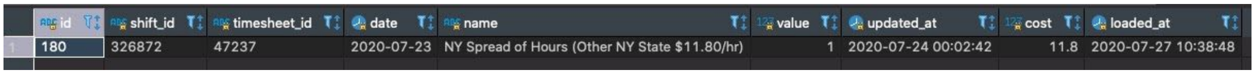
The resulting tables should look like this:

shift_breaks (for shift_id ‘326872’):



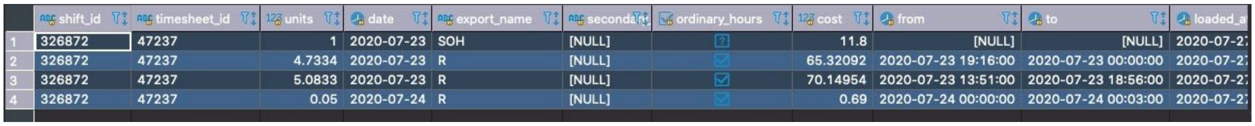
	id	shift_id	timesheet_id	date	start	finish	length	paid	updated_at	loaded_at
1	96575	326872	47237	2020-07-23	2020-07-23 18:56:00	2020-07-23 19:16:00	20	□	2020-07-23 19:16:14	2020-07-27 10:38:40

shift_allowances (for shift_id ‘326872’):



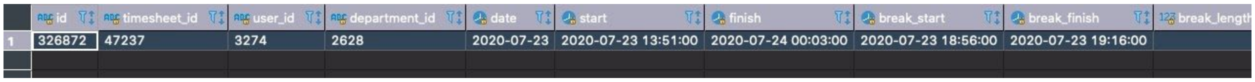
	id	shift_id	timesheet_id	date	name	value	updated_at	cost	loaded_at
1	180	326872	47237	2020-07-23	NY Spread of Hours (Other NY State $11.80/hr)	1	2020-07-24 00:02:42	11.8	2020-07-27 10:38:48

shift_award_interpretation (for shift_id ‘326872’):

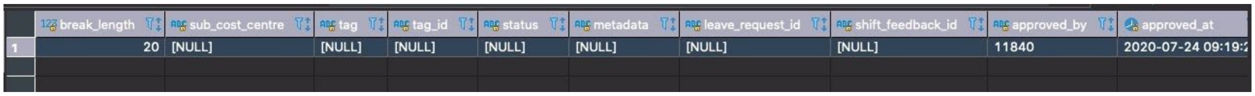


	shift_id	timesheet_id	units	date	export_name	seconda	ordinary_hours	cost	from	to	loaded_a
1	326872	47237	1	2020-07-23	SOH	[NULL]	?	11.8	[NULL]	[NULL]	2020-07-2
2	326872	47237	4.7334	2020-07-23	R	[NULL]	☑	65.32092	2020-07-23 19:16:00	2020-07-23 00:00:00	2020-07-2
3	326872	47237	5.0833	2020-07-23	R	[NULL]	☑	70.14954	2020-07-23 13:51:00	2020-07-23 18:56:00	2020-07-2
4	326872	47237	0.05	2020-07-24	R	[NULL]	☑	0.69	2020-07-24 00:00:00	2020-07-24 00:03:00	2020-07-2

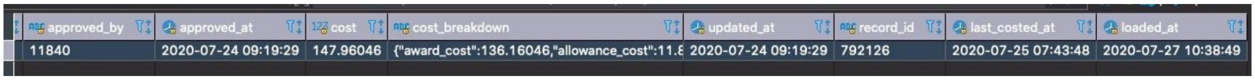
shifts (for shift_id ‘326872’, in three screenshots because there are a lot of columns):



	id	timesheet_id	user_id	department_id	date	start	finish	break_start	break_finish	break_length
1	326872	47237	3274	2628	2020-07-23	2020-07-23 13:51:00	2020-07-24 00:03:00	2020-07-23 18:56:00	2020-07-23 19:16:00	



	break_length	sub_cost_centre	tag	tag_id	status	metadata	leave_request_id	shift_feedback_id	approved_by	approved_at
1	20	[NULL]	[NULL]	[NULL]	[NULL]	[NULL]	[NULL]	[NULL]	11840	2020-07-24 09:19:2



approved_by	approved_at	cost	cost_breakdown	updated_at	record_id	last_costed_at	loaded_at
11840	2020-07-24 09:19:29	147.96046	{"award_cost":136.16046,"allowance_cost":11.8	2020-07-24 09:19:29	792126	2020-07-25 07:43:48	2020-07-27 10:38:49