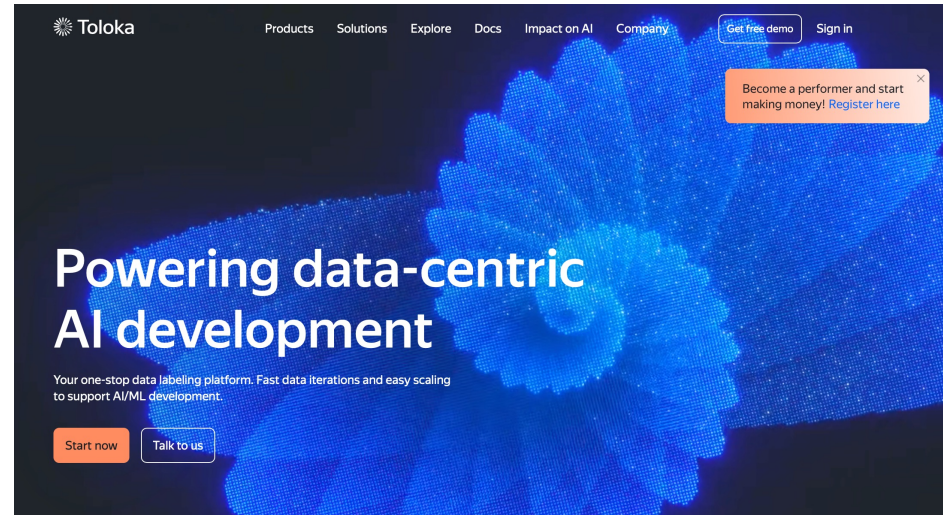




Website Relevance Step-by-step guide

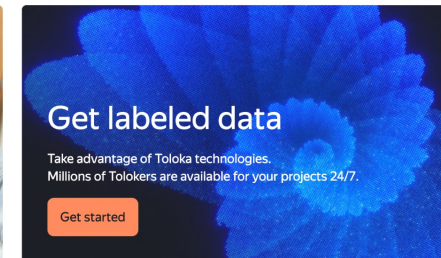
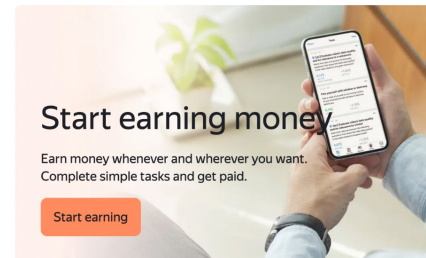
Apply promo code

1. Go to toloka.ai and click on **Start now**

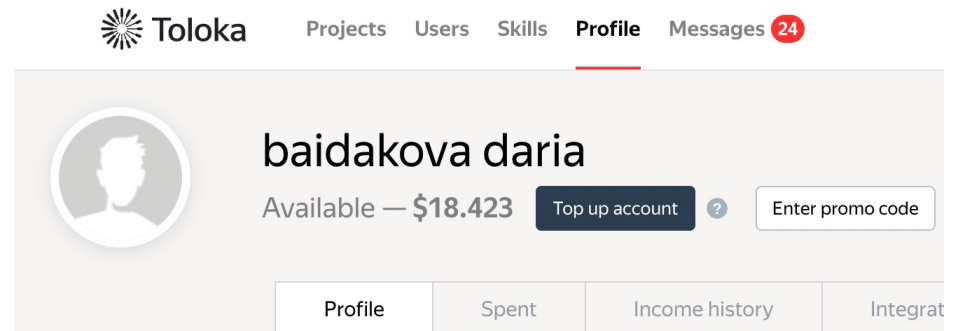


2. Click on **Get labeled data (blue)** and use your email and password to register

I want to...

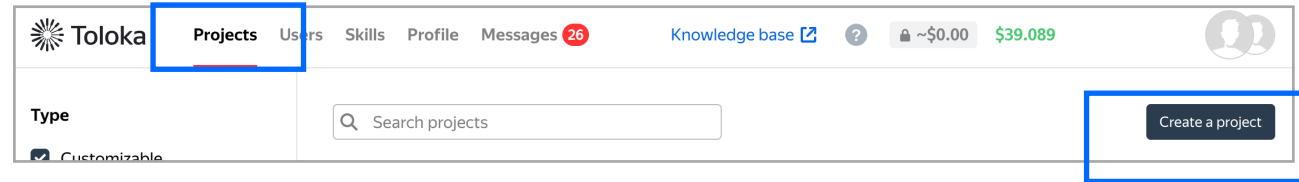


3. Go to **Profile section** and Click the button **Enter promo code**
[TOLOKA_ICWE2022](#)



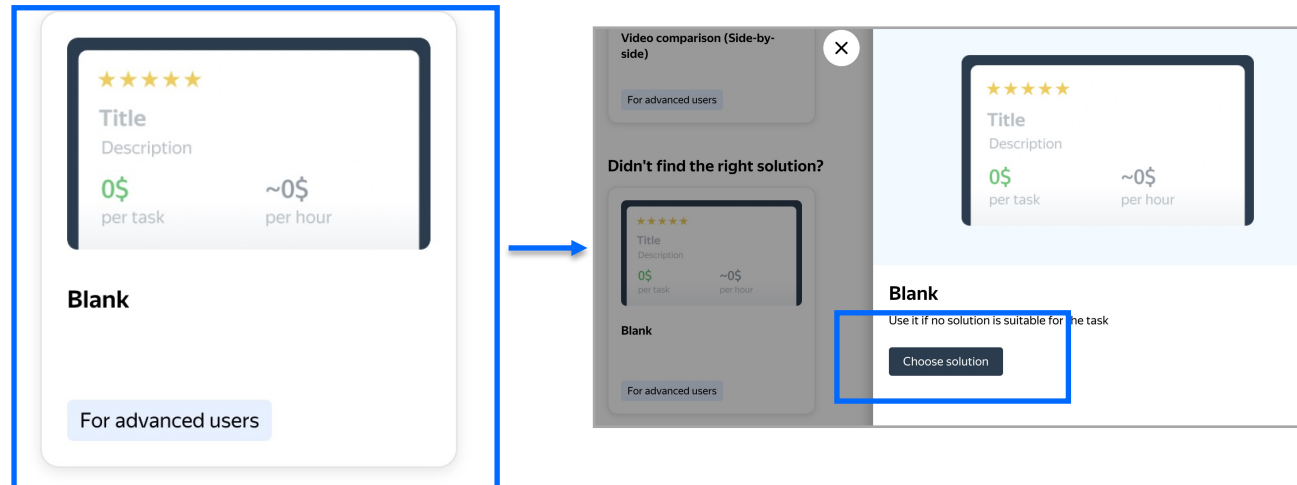
Create a project

1. Go to **Projects** and Click the button + **Create project**



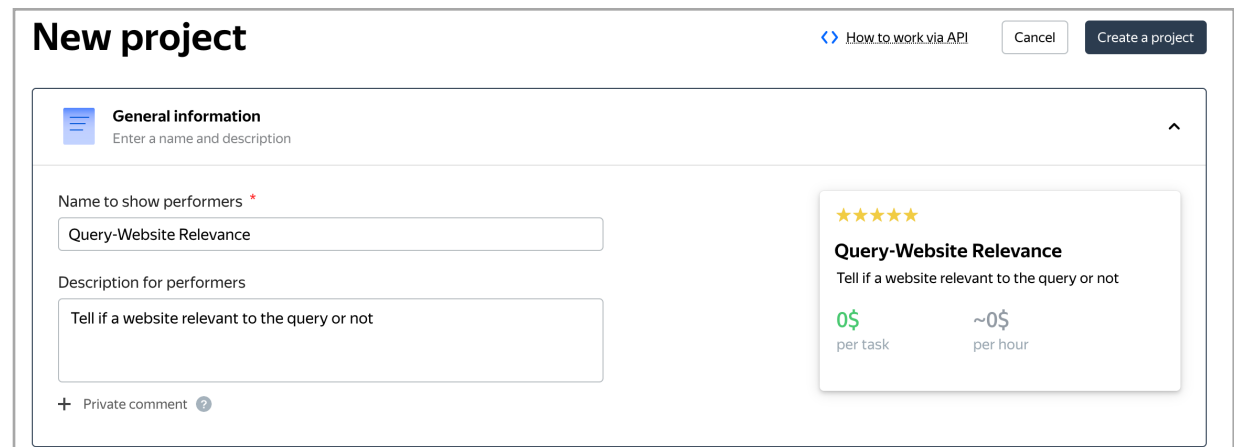
2. Choose the **blank** template and click the button + **Choose solution**

Didn't find the right solution?




3. Enter a clear project name and description in General information

Important: It will be visible to others



4. Update the task interface in the Template Builder block

 **Task interface**
Create a user-friendly interface for performers

Editor

HTML / JS / CSS ? **Template builder** ?

- 4.1. Delete all code you see there and replace it with [another](#) one in **Config** section

Note: you can also find new code in appendix

New project

[How to work via API](#) Cancel Create a project

Config

```
1 | "view": {
2 |   "type": "view.list",
3 |   "items": [
4 |     {
5 |       "type": "view.link",
6 |       "content": {
7 |         "type": "data.input",
8 |         "path": "query"
9 |       },
10 |     },
11 |     {
12 |       "type": "helper.search-query",
13 |       "query": {
14 |         "type": "data.input",
15 |         "path": "query"
16 |       },
17 |       "engine": "google"
18 |     }
19 |   ]
20 | },
21 | {
22 |   "type": "view.image",
23 |   "maxWidth": 800,
24 |   "scrollable": true,
25 |   "url": {
26 |     "type": "data.input",
27 |     "path": "image"
28 |   }
29 | },
30 | {
31 |   "type": "field.radio-group",
32 |   "validation": {
33 |     "type": "condition.required"
```

Preview

Search query
<https://www.google.ru/search?q=undefined>

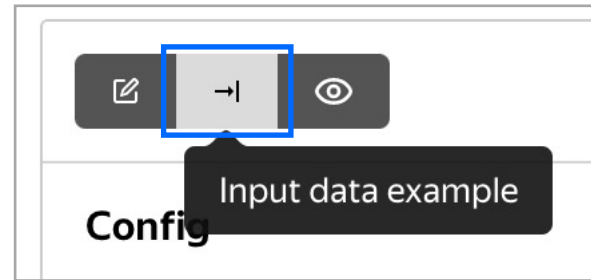
Choose category

1 Relevant ⓘ

2 Irrelevant ⓘ

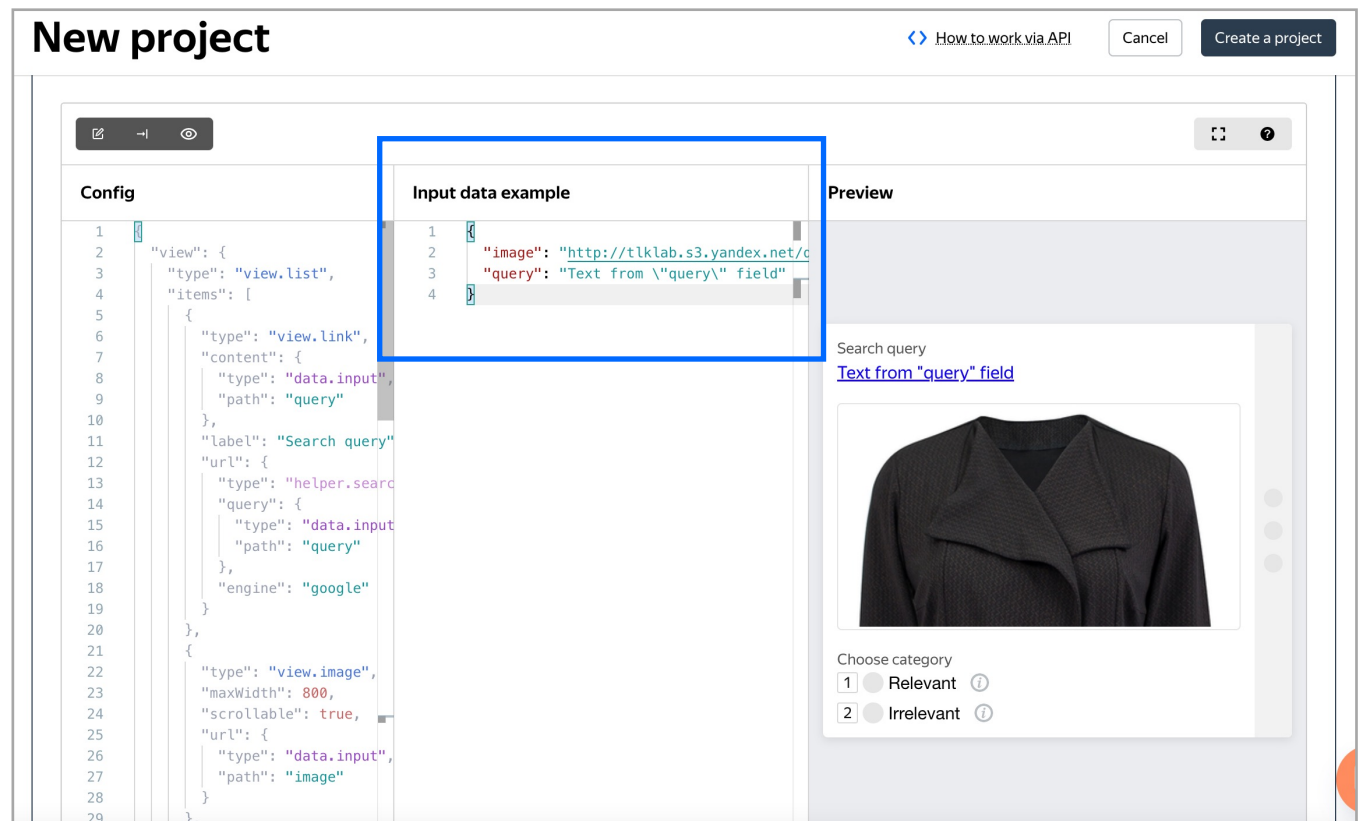
Submit Reset

4.2. Now you need to change input data so open this extra field



4.2. Delete all code you see there and replace it with [another](#) one

Note: you can also find new code in appendix



4.3. Leave data specification settings at default.

Data specification ?

Input data

- query (string) ●
- image (URL) ●

Output data <>

- relevance (string) ●

Define data specification manually
If the generated specification doesn't work for your task, define custom input and output data. Automatic update of specs due to task interface changes will be disabled. [Learn more](#)

Show common interface elements

5. Write your own short and simple instructions.

Or you can copy and paste our [instructions](#)

6. Click **Create a project** to save the project

7.**Note.** To edit project parameters, click the button in the list of projects

Create a training pool

1. Click on the **Training** tab and then click **Add training**.

Note: Training is an essential part of almost every crowdsourcing project. It allows you to select annotators who have really mastered the task, and thus improve quality. Training is also a great tool for scaling your task because you can run it any time you need new annotators.

The screenshot shows the Toloka project management interface for a project titled "Query-Website Relevance" (ID 99427). The project is currently "Active". The interface includes a navigation menu with tabs for "Pools", "Training", "Statistics", and "Quality control". The "Training" tab is selected and highlighted with a blue box. Below the navigation, there are buttons for "Active and closed", "Archived", "Filters", and a search input field. A prominent "Add training" button is highlighted with a blue box in the top right corner. Below these elements, there is a table with columns for "Title", "Completed", "Status", "Started", and "To be completed". A note states: "Pools can be archived manually or automatically (automatic archiving applies to pools with no activity for 30 days)". At the bottom right, there is a dropdown menu showing "50".

2. Use the existing project instructions.

The screenshot shows the "Training" configuration page for the "Query-Website Relevance" project. The page title is "Training". Under the "Instructions" section, there is a checkbox labeled "Use project instructions" which is checked and highlighted with a blue box. Below this, there is a text area containing the instruction: "In this task you need to evaluate the **relevance of webpages in the context of commercial queries.**". The page also includes a "About the task" section which is partially visible at the bottom.

3. Specify the training pool settings:

3.1 And then click **Create training**.

General settings

Training title: Search website relevance

Price per task: free training

Adult content: No

Time per task suite: 600 seconds

Retry after: days

Task assignment settings

Assign in order of uploading

Shuffle on page

Settings for passing training

Full completion: Yes

Required to pass: Number of pages

Create training

4. Once the pool is created you can upload tasks. Press Upload. You can upload the tasks with comments straight from the file as in our example which is [here](#).

Note: It's important to include examples for all classes in the training. When running your own projects make sure the training set is balanced and the comments explain why an answer is correct. Don't just name the correct answers.

Search website relevance — closed

Statistics Download results Edit

Download the sample file, add your task data, and upload the file to the pool. The sample file uses TSV format, which is the same as CSV but which uses a tab as the separator. Make sure you choose UTF-8 encoding when saving the file. [Learn more in the Guide](#)

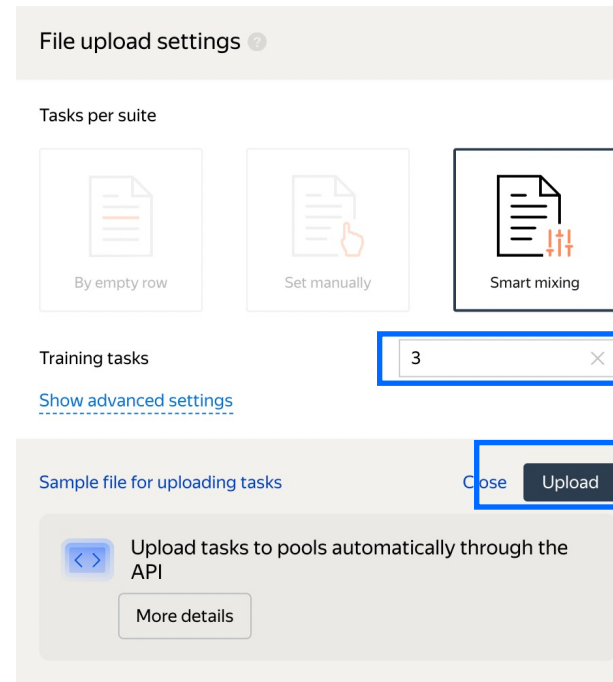
- Template for general tasks.tsv
- Template for control tasks.tsv
- Template for training tasks.tsv

Upload

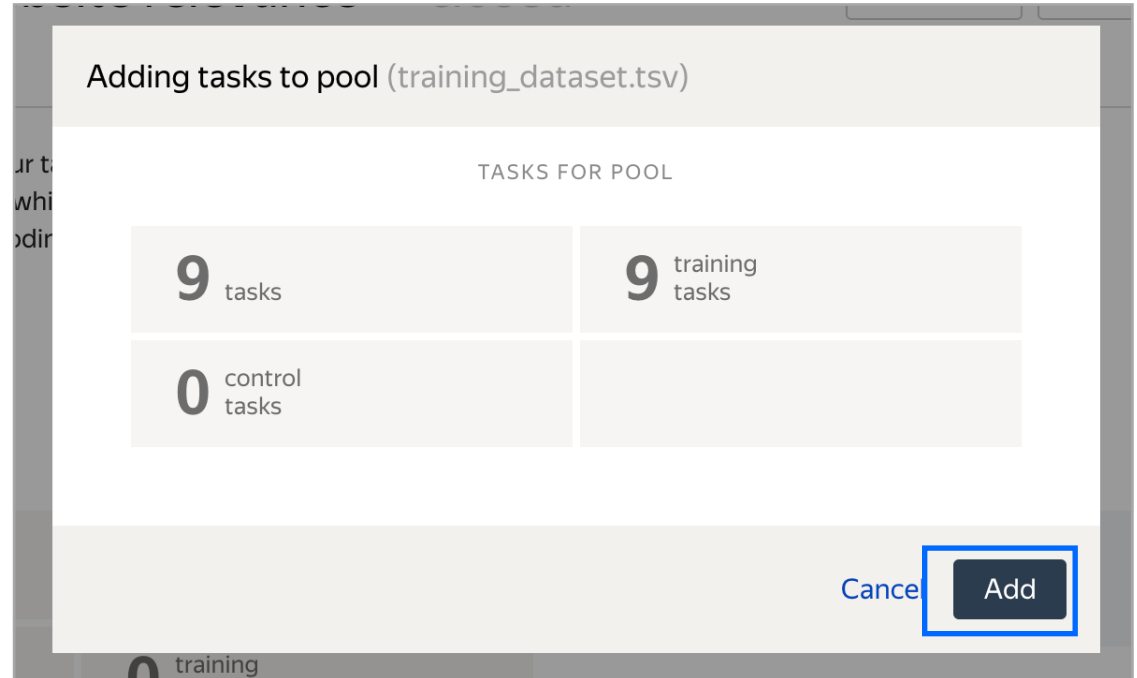
0 task pages	0 training tasks
0 tasks	0 control tasks

0 Users who completed training

5. In the opened window choose **Smart mixing**. It is possible to make 1 or several task pages. As we have 9 training tasks it seems just reasonable to make 3 task pages for an annotator to learn.



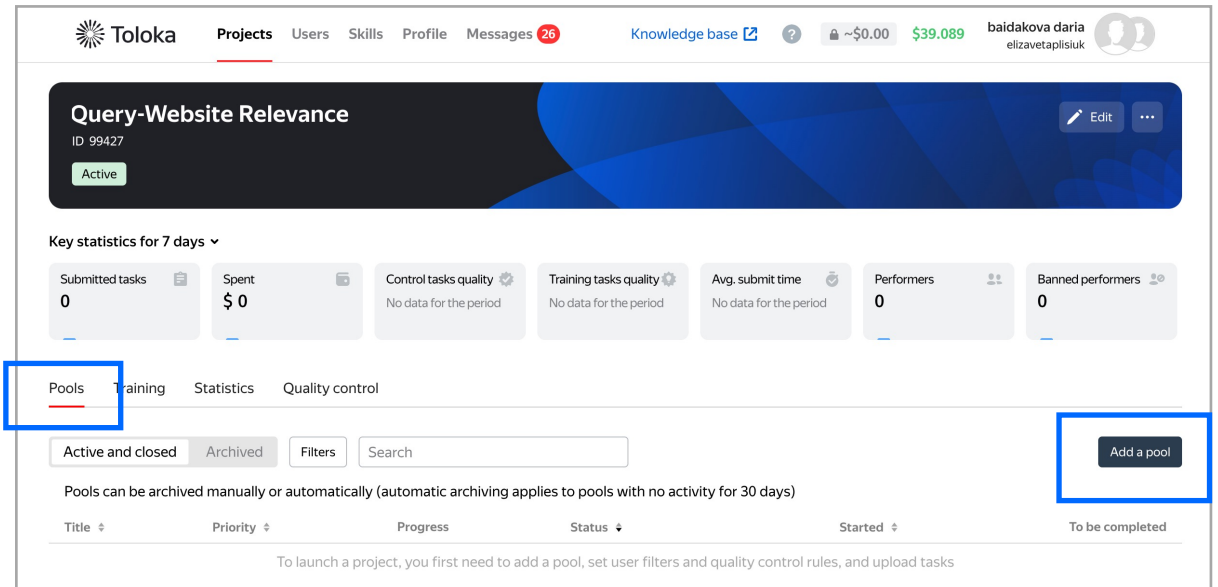
6. After the file has been uploaded press **Add** to complete the pool.



Create a main pool

1. Now go back to our project and click on **Pools** and then **add a pool**

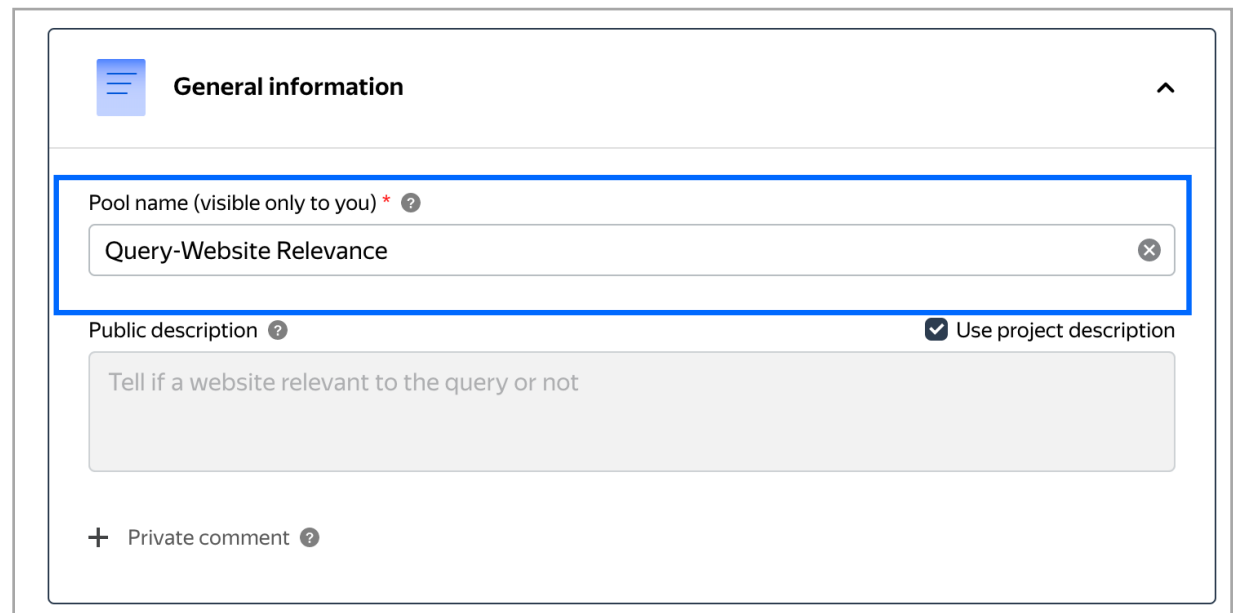
Note: to get back to the project page use these transitions Projects > Project_name



The screenshot shows the Toloka project interface for 'Query-Website Relevance' (ID 99427). The 'Pools' tab is selected and highlighted with a blue box. Below the tabs, there are filters for 'Active and closed' and 'Archived', and a search bar. A blue box highlights the 'Add a pool' button in the top right corner. The page also displays key statistics for 7 days, including Submitted tasks (0), Spent (\$0), and Performers (0).

2. Give the pool any convenient name.
You are the only one who will see it.

There are two types of the description: public and private. Choose the option you prefer



The screenshot shows the 'General information' form for creating a pool. The 'Pool name (visible only to you)' field is highlighted with a blue box and contains the text 'Query-Website Relevance'. Below it, the 'Public description' section is visible, with a checkbox for 'Use project description' checked. The public description text area contains the placeholder text 'Tell if a website relevant to the query or not'. There is also a '+ Private comment' field at the bottom.

3. Now we need to filter an **audience**.

Filter annotators who can access the task. Make sure to uncheck the option 'My tasks may contain shocking ...' if it does not.

Choose the **Languages** options in the list.

Specify the percentage of top-rated annotators in the Speed / quality balance.

Audience

Filter your audience by language or country. Otherwise, performers anywhere in the world can access your tasks. You can [copy audience filters and quality control settings](#) from another pool. [Learn more](#)

My tasks may contain shocking or pornographic content. [Learn more](#)

Languages Performers who passed the language test [?](#)

+ Add filter + Add skill

Speed/quality balance
Note that fewer users means slower pool completion
[Learn more](#)

Top % Online

Specify the percentage of top-rated users who can access tasks in the pool

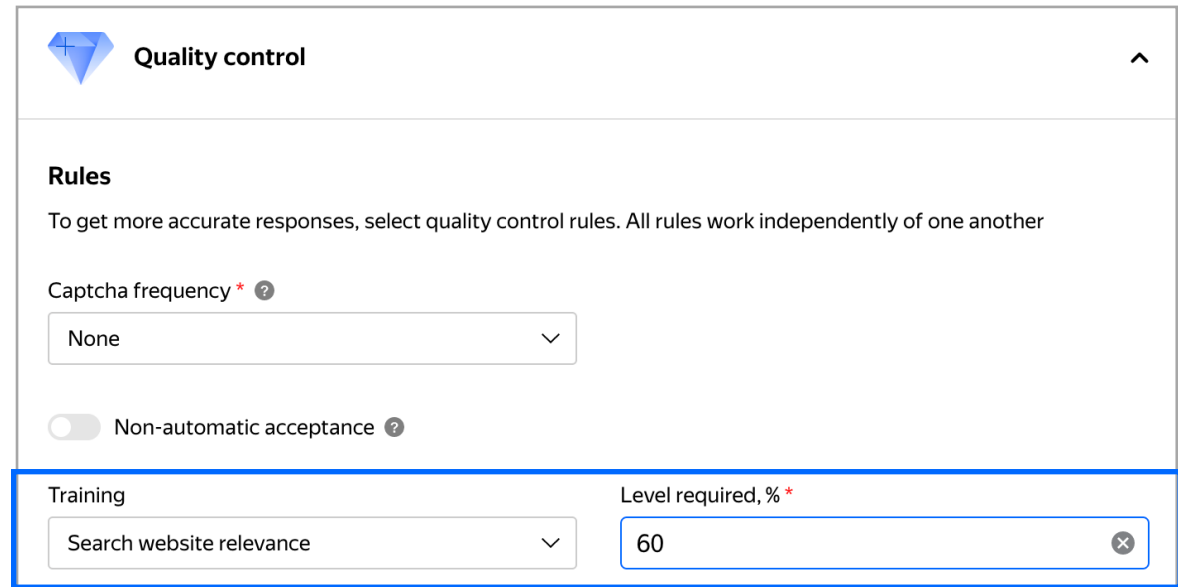
9929 Speed 100% 90% 80% 70% 60% 50% 40% 30% 20% 10% 992 Quality

80% of the best performers were selected
The task is available to **7943** active users

4. Set up **Quality control**.

Attach the training you created earlier and select the accuracy level that is required to reach the main pool.

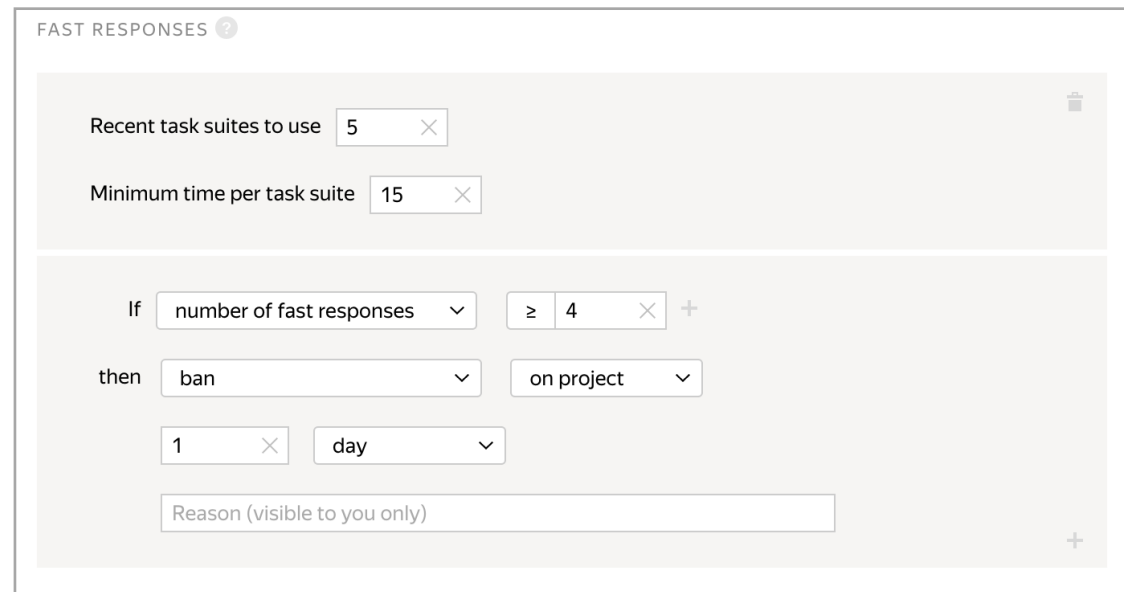
Note: This means that annotators who get less than 60% accuracy will not see this pool. This percentage means that the annotators accepted to the main pool have given at least 5 correct answers in the training pool.



The screenshot shows the 'Quality control' settings page. At the top, there is a blue diamond icon and the title 'Quality control'. Below this, the 'Rules' section is titled 'Rules' and includes the instruction: 'To get more accurate responses, select quality control rules. All rules work independently of one another'. There are two main settings: 'Captcha frequency' set to 'None' and 'Non-automatic acceptance' which is turned off. A blue box highlights the 'Training' dropdown set to 'Search website relevance' and the 'Level required, %' dropdown set to '60'.

5. Leave the **Fast responses** rule as it is

Note: This rule allows you to ban annotators who submit tasks at a suspiciously high speed.



The screenshot shows the configuration for the 'FAST RESPONSES' rule. It includes a title 'FAST RESPONSES' with a help icon. The configuration is as follows: 'Recent task suites to use' is set to 5; 'Minimum time per task suite' is set to 15. The rule logic is: 'If number of fast responses ≥ 4, then ban on project for 1 day'. There is a text field for 'Reason (visible to you only)'.

6. Leave the **Majority vote** rule as it is

MAJORITY VOTE ?

Accept as majority

Recent tasks to use

If +

then

7. Let's add one more role: **control tasks**

Click **Add a quality control rule** and choose **Control tasks**.

Rules

dishonest performers

Submitted responses
Limits assignments per performer. This gives you a broader selection of performers

Recompletion of assignments from banned users
If a performer is banned, their completed assignments are reassigned to other performers

Processing rejected and accepted assignments
If an assignment was rejected, it is reassigned to other performers

Control tasks
Tracks responses to control tasks. This helps to identify high-quality performers and ban those who often make mistakes

Skipped assignments
Limits the number of task suites that can be skipped in a row

+ Add a quality control rule

1

2

8. Set the number of responses and the percentage of correct responses. Ban annotators who give incorrect responses to control tasks.

Note: Since the projects such as this one can have an answer that can be used as ground truth, we can use standard quality control rules like control tasks.

CONTROL TASKS ?

Recent control task responses to use

If +

and +

then

9. Set the **price** per task suite (for example, \$0.05).

Also, choose an **overlap** of 3.

Click **Create a pool**

Price

Price per task suite, \$ * ?

Set at least \$0.02 for simple tasks
Set at least \$0.05 for complex tasks

Performer interest at this price ? **Medium** 📊

Recommended number of tasks per suite ? **10**

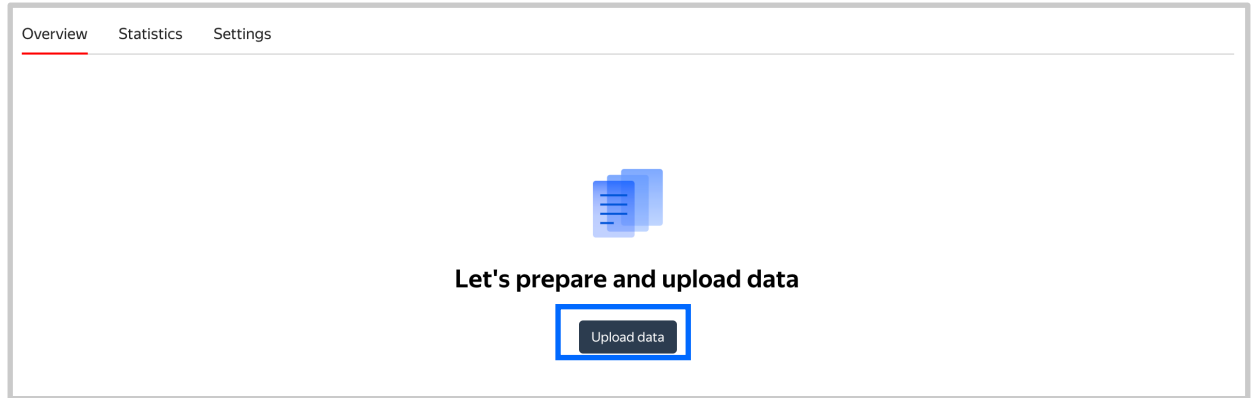
Overlap * ?

For simple tasks recommended overlap is 3

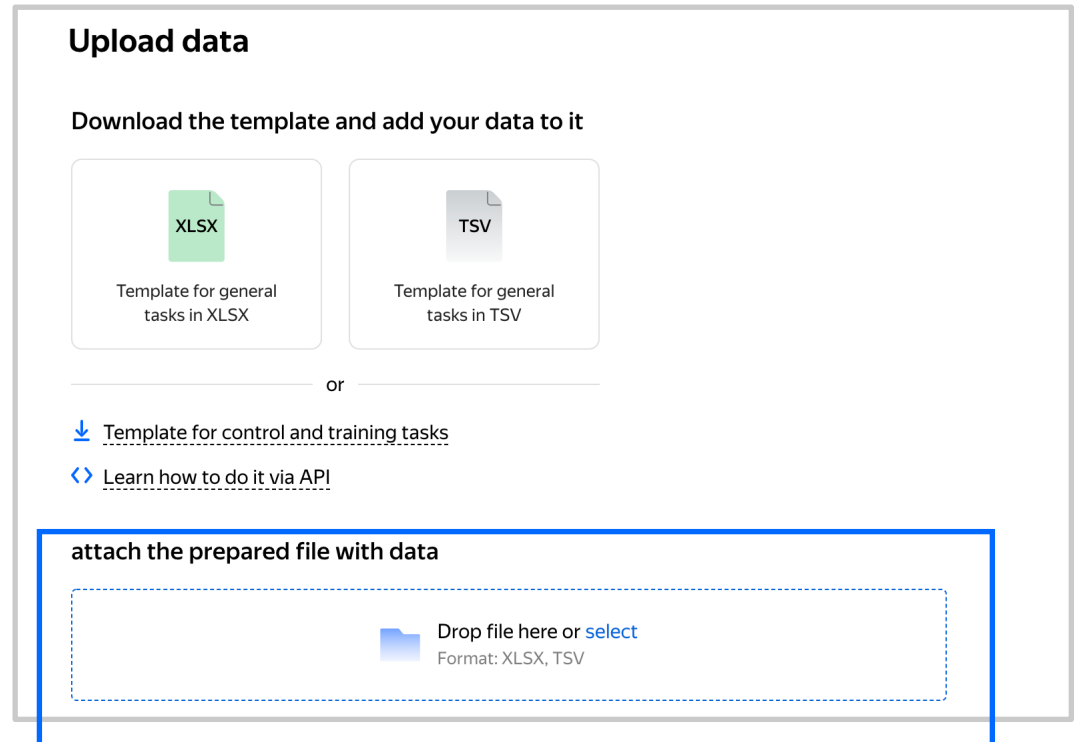
Price per 1 task **\$0.008**
Including 30% fee

Prepare and upload a dataset

1. Upload pool task from [this file](#): click on the button “Upload data”



Now drop the dataset on the selected area



2. Select **Smart mixing** in File upload settings and specify the number of tasks of each type per page. Click **combine tasks into suites**.

We recommend putting as many tasks on one page as an annotator can complete in 1 to 5 minutes. That way, annotators are less likely to get tired, and they won't lose a significant amount of data if a technical issue occurs.

How many tasks do you want per suite?

Performers get **0.05 \$** per task suite

Smart mixing Set manually

Set how many tasks of each type to mix in each task suite. For better quality, include control tasks. You can mark up control tasks on the pool page after uploading data, or upload them in a separate file and adjust the smart mixing settings later. [Learn more](#)

Number of general tasks * ?	Number of training tasks * ?	Number of control tasks * ?
<input type="text" value="8"/>	<input type="text" value="0"/>	<input type="text" value="2"/>

Assign partial page ?

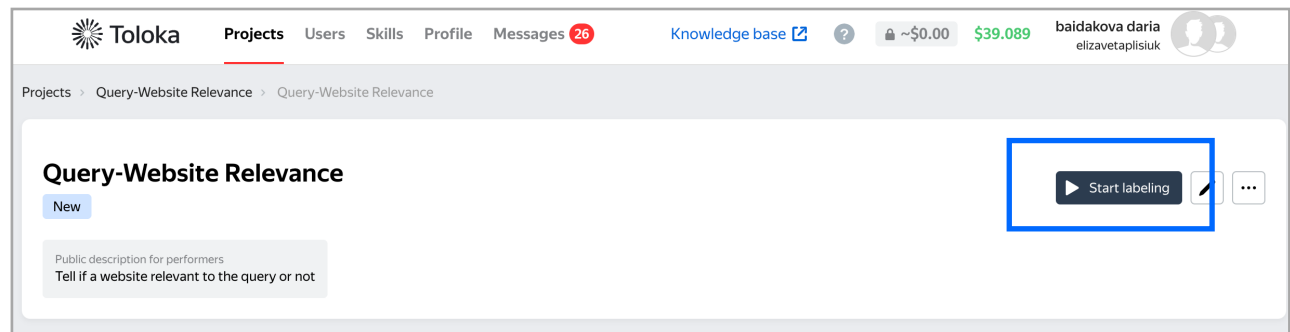
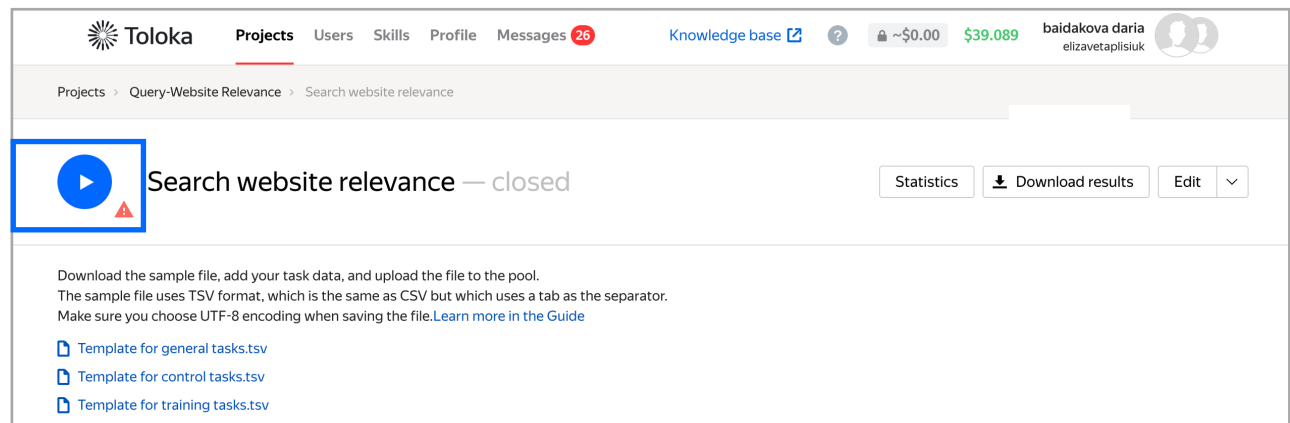
Minimum general tasks ?	Minimum training tasks ?	Minimum control tasks ?
<input type="text"/>	<input type="text"/>	<input type="text"/>

Combine tasks into suites Cancel

3. If everything is okey, you will receive this message



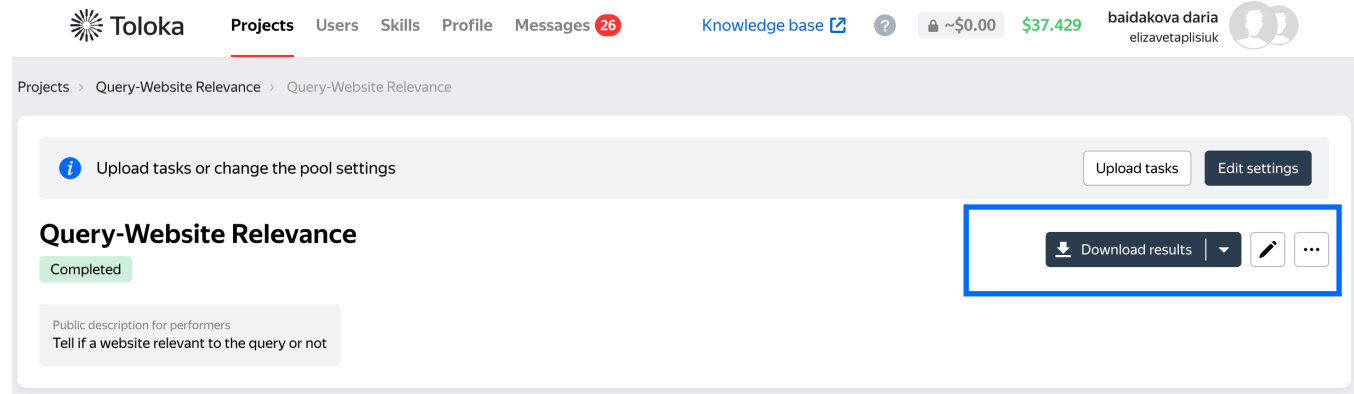
4.1. Start your training and main pools.



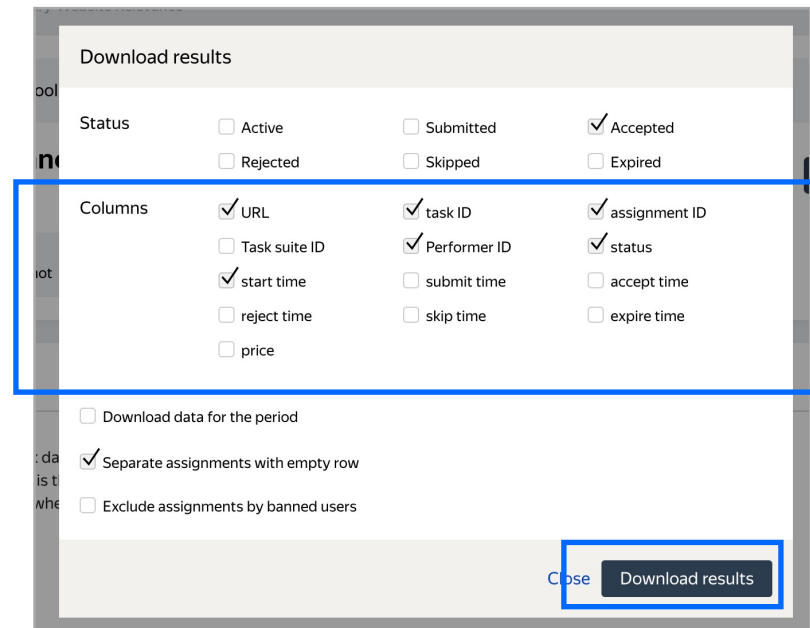
Download results

1. Wait until the main pool is completed.

2. Click **Download results**



3. Choose columns you need and click **download results**. We will aggregate results together in the next part of the tutorial.



Appendix

Interface code Step 4.1.

```
{
  "view": {
    "type": "view.list",
    "items": [
      {
        "type": "view.link",
        "content": {
          "type": "data.input",
          "path": "query"
        },
        "label": "Search query",
        "url": {
          "type": "helper.search-query",
          "query": {
            "type": "data.input",
            "path": "query"
          },
          "engine": "google"
        }
      },
      {
        "type": "view.image",
        "maxWidth": 800,
        "scrollable": true,
        "url": {
          "type": "data.input",
          "path": "image"
        }
      }
    ],
    "type": "field.radio-group",
    "validation": {
      "type": "condition.required"
    },
    "label": "Choose category",
    "options": [
      {
        "label": "Relevant",
        "value": "RELEVANT",
        "hint": "document corresponds to a query, provides the requested information and can be indirectly related to the query"
      },
      {
        "label": "Irrelevant",
        "value": "IRRELEVANT",
        "hint": "page doesn't contain the object of the query at all or its part is insufficient in comparison with the main contents of the page (so-called \"random entry\")"
      }
    ],
    "data": {
      "type": "data.output",
      "path": "relevance"
    }
  }
}
}
}
"plugins": [
  {
    "1": {
      "type": "action.set",
      "data": {
        "type": "data.output",
        "path": "relevance"
      },
      "payload": "RELEVANT"
    },
    "2": {
      "type": "action.set",
      "data": {
        "type": "data.output",
        "path": "relevance"
      },
      "payload": "IRRELEVANT"
    },
    "type": "plugin.hotkeys"
  },
  {
    "type": "plugin.toloka",
    "layout": {
      "kind": "scroll",
      "taskWidth": 800
    }
  }
]
}
```

Input data code Step 4.2.

```
{  
  "image": "http://tlklab.s3.yandex.net/demo_1/32c8b44d-cc9f-41b5-85aa-98096774926b",  
  "query": "Text from \"query\" field"  
}
```

Useful links:

1. Config https://toloka.ai/files/icwe_2022/tb_config.txt
2. Input data https://toloka.ai/files/icwe_2022/tb_input_data.txt
3. Instructions https://toloka.ai/files/icwe_2022/instructions.txt
4. Train dataset https://toloka.ai/files/icwe_2022/train_dataset.tsv
5. Main dataset https://toloka.ai/files/icwe_2022/main_dataset.tsv