

Генерация приветствий

 **Endpoint:**

<https://functions.yandexcloud.net/d4ekvkrc5sl885id7g3i>

 **Метод:** `POST`

 **Content-Type:** `application/json`

Request Body

```
{
  "user_id": "string",           // ID пользователя
  "match_profile_description": "string", // Описание профиля мэтча
  "match_gender": "male" | "female",    // Пол мэтча
  "saved_greetings": ["string", ...],    // (опц.) сохранённые шаблоны
  "past_messages": ["string", ...]      // (опц.) прошлые сообщения
}
```

! Пол пользователя автоматически определяется как противоположный полу мэтча.

Response

```
{
  "top_saved_greetings": ["string", "string"], // 2 релевантных шаблона
  "generated_formal": "string",                // Вежливое, формальное приветствие
  "generated_informal": "string"              // Лёгкое, неформальное приветствие
}
```

 **Пример запроса через** `curl`

```
curl -X POST https://functions.yandexcloud.net/d4ekvkrc5sl885id7g3i \
-H "Content-Type: application/json" \
-d '{
  "user_id": "user123",
  "match_profile_description": "Обожаю кататься на велосипеде и пить кофе в центре",
  "match_gender": "female",
  "saved_greetings": ["Привет! Чем увлекаешься?", "Хей! Расскажи, как прошёл день :)"],
  "past_messages": ["Привет, я часто бываю в кофейнях"]
}'
```

Пример ответа

```
{
  "top_saved_greetings": [
    "Привет! Чем увлекаешься?",
    "Хей! Расскажи, как прошёл день :)"
  ],
  "generated_formal": "Привет! Увидел, что ты любишь кататься на велосипеде. Это здорово! Какое место в центре твоё любимое для кофе?",
  "generated_informal": "Эй! Велосипеды и кофе — классная комбо. Когда поедем кататься вместе? 😊"
}
```

Код:

```
import os
import json
from typing import List, Dict
from yandex_cloud_ml_sdk import YCloudML

sdk = YCloudML(folder_id=YC_FOLDER_ID, auth=YC_API_KEY)
```

```

def select_best_saved(saved: List[str], profile: str, match_gender: str, user_gender: str):
    if len(saved) <= 2:
        return saved
    prompt = (
        "Ты — эксперт в онлайн-знакомствах. Мэтч:\n"
        f"{profile}\nПол мэтча: {match_gender}. Пол юзера: {user_gender}.\n"
        "Среди этих шаблонов выбери 2 лучших:\n" +
        "\n".join(f"{i+1}. {s}" for i, s in enumerate(saved)) +
        "\nВерни JSON массив из двух строк."
    )
    res = sdk.models.completions("yandexgpt-lite").configure(temperature=0.2).run(prompt)
    try:
        sel = json.loads(res[0].text.strip())
        return [s for s in sel if s in saved][:2]
    except:
        return saved[:2]

def ask(prompt: str) -> str:
    res = sdk.models.completions("yandexgpt-lite").configure(temperature=0.8).run(prompt)
    return res[0].text.strip()

def generate_greeting(body: Dict) -> Dict:
    profile = body["match_profile_description"]
    match_gender = body["match_gender"]
    saved = body.get("saved_greetings") or []
    past = body.get("past_messages") or []
    history = "\n".join(saved + past)
    user_gender = "male" if match_gender == "female" else "female"

    top = select_best_saved(saved, profile, match_gender, user_gender)

    formal = ask(
        f"Ты вежливый пользователь.\nПол мэтча: {match_gender}, юзера: {user_gender}.\n"
        f"Профиль: {profile}\nИстория:\n{history}\n"
        "Сформулируй вежливое формальное приветствие."
    )

```

```
informal = ask(
    f"Ты открытый и лёгкий пользователь.\nПол мэтча: {match_gender}, юзе
    f"Профиль: {profile}\nИстория:\n{history}\n"
    "Сформулируй непринуждённое, дружелюбное приветствие."
)

return {
    "top_saved_greetings": top,
    "generated_formal": formal,
    "generated_informal": informal
}
```