

```
In [2]: %load_ext autoreload
        %autoreload 2
```

```
In [1]: import calendar
        from collections import Counter
        from functools import reduce
        from operator import itemgetter
        from functools import partial

        import pandas as pd
        import plotly.express as px
        import plotly.figure_factory as ff
        import plotly.offline as pyo
        from mongoengine import connect

        from src import settings
        from src.data.vacancy import Vacancy
        from src.features.clean import remove_html
        from src.visualization.statistics import plot_value_counts
```

```
In [3]: connect(
        host=settings.db_host,
        port=settings.db_port,
        db=settings.db_name
    )
```

```
Out[3]: MongoClient(host=['localhost:27017'], document_class=dict, tz_aware=False, connect=True,
read_preference=Primary())
```

```
In [5]: pyo.init_notebook_mode()
```

```
In [6]: df: pd.DataFrame = (
        Vacancy
        .objects
        .to_dataframe(include=[
            '_id',
            'name',
            'description',
            'salary',
            'schedule.name',
            'experience',
            'employment.name',
            'area.name',
            'address.lat',
            'address.lng',
            'address.city',
            'published_at',
            'specializations',
            'employer.name',
            'professional_roles',
            'key_skills',
        ])
    )
```

```
In [7]: df.set_index('_id', inplace=True)
```

```
In [8]: df['description'] = df['description'].map(remove_html)
```

```
In [9]: df.columns
```

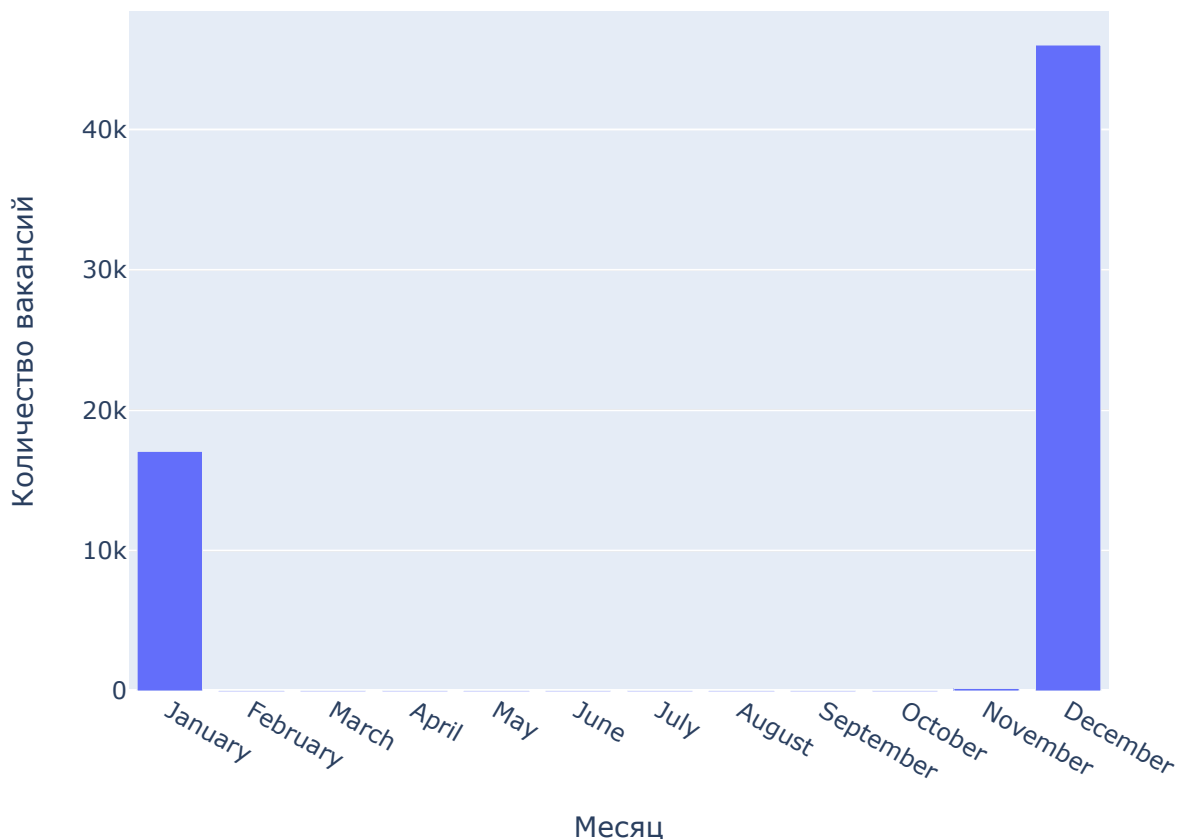
```
Out[9]: Index(['description', 'key_skills', 'schedule.name', 'experience.id',  
        'experience.name', 'employment.name', 'salary.to', 'salary.from',  
        'salary.currency', 'salary.gross', 'name', 'area.name', 'published_at',  
        'employer.name', 'specializations', 'professional_roles',  
        'address.city', 'address.lat', 'address.lng'],  
        dtype='object')
```

```
In [10]: df.shape
```

```
Out[10]: (63273, 19)
```

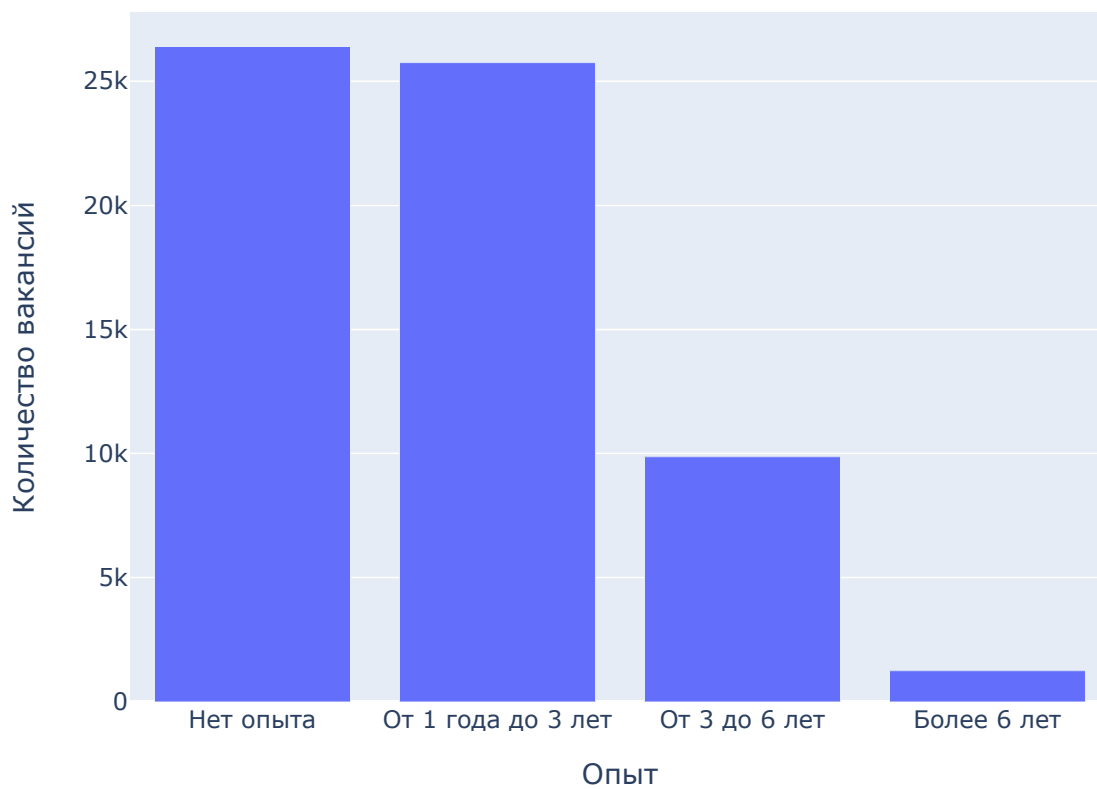
```
In [11]: df.published_at = pd.to_datetime(df.published_at)  
  
count_by_month = {  
    calendar.month_name[month]: sum(df.published_at.dt.month == month) for month in range(1, 13)  
}  
  
px.bar(  
    x=count_by_month.keys(),  
    y=count_by_month.values(),  
    labels={'x': 'Месяц', 'y': 'Количество вакансий'},  
    title='Количество вакансий в зависимости от месяца'  
)
```

## Количество вакансий в зависимости от месяца



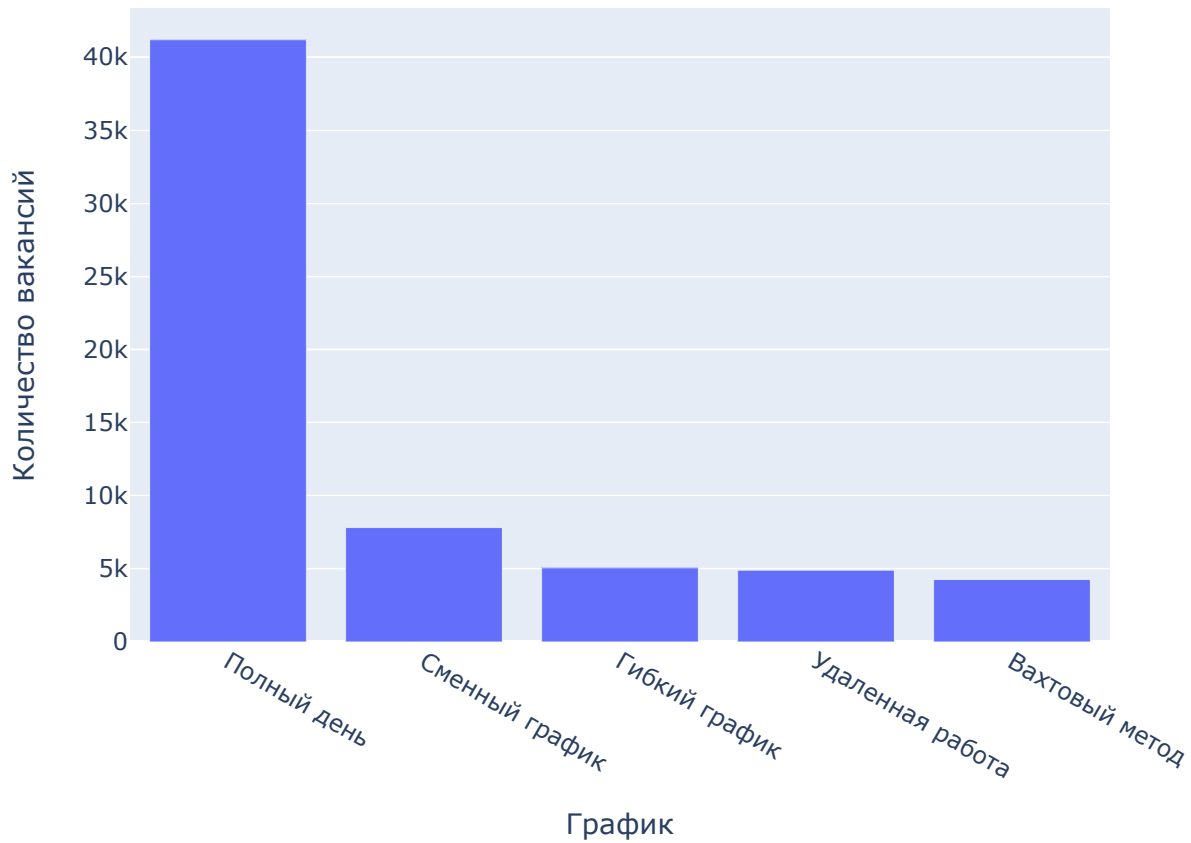
```
In [12]: plot_value_counts(  
    df['experience.name'],  
    x_label='Опыт',  
    y_label='Количество вакансий',  
    title='Количество вакансий в зависимости от опыта'  
).update_xaxes(categoryorder='total descending')
```

### Количество вакансий в зависимости от опыта



```
In [13]: plot_value_counts(  
    df['schedule.name'],  
    x_label='График',  
    y_label='Количество вакансий',  
    title='Количество вакансий в зависимости от графика работы'  
).update_xaxes(categoryorder='total descending')
```

## Количество вакансий в зависимости от графика работы



## Анализ навыков

```
In [14]: key_skills = reduce(set.union, df.key_skills, set())
```

```
In [15]: len(key_skills)
```

```
Out[15]: 10040
```

```
In [16]: count_by_key_skill = reduce(Counter.__add__, map(Counter, df.key_skills))
```

```
In [17]: ff.create_table(['Навык', 'Количество вакансий']) + count_by_key_skill.most_common(50)
```

Навык	Количество вакансий
Работа в команде	13036
Пользователь ПК	9466
Грамотная речь	9344
Умение работать в команде	4201
Деловое общение	2895
Активные продажи	2801
Организаторские навыки	2480
Навыки продаж	2411
Английский язык	2343
Ведение переговоров	2232
Телефонные переговоры	2027
Грамотность	2023
Деловая переписка	1934
Обучение и развитие	1725
Работа с большим объемом информации	1649
Работа в условиях многозадачности	1596
Управление персоналом	1571
Ориентация на результат	1378
Клиентоориентированность	1369
Поиск и привлечение клиентов	1355
Работа с людьми	1235
Точность и внимательность к деталям	1204
Водительское удостоверение категории В	1155
Заключение договоров	1125
Прямые продажи	1098
Обучение персонала	1089
Умение работать в коллективе	1072
Поиск информации в интернет	1052
Деловая коммуникация	1044
AutoCAD	1039
MS PowerPoint	939
Управление проектами	920
оформление документации	870

Консультирование	852
Навыки межличностного общения	842
1С: Предприятие 8	841
Контроль качества	802
Подбор персонала	796
SQL	793
Ведение документации	792
Материальная ответственность	779
Знание устройства автомобиля	764
Работа с кассой	746
Развитие продаж	728
Техническое обслуживание	719
Креативность	711
CRM	702
Мобильность	697
Ведение отчетности	685
Холодные продажи	668

## Анализ профобластей

```
In [18]: df['profarea_names'] = df.specializations.map(lambda specs: list(set(map(itemgetter('pr
```

```
In [19]: df.profarea_names.head(10)
```

```
Out[19]: _id
49810439      [Транспорт, логистика]
49810551      [Административный персонал, Домашний персонал]
49810468      [Спортивные клубы, фитнес, салоны красоты]
45788942      [Транспорт, логистика]
49810601      [Бухгалтерия, управленческий учет, финансы пре...
49810507      [Административный персонал]
49810469      [Спортивные клубы, фитнес, салоны красоты]
49810426      [Спортивные клубы, фитнес, салоны красоты]
47003369      [Продажи]
49810583      [Производство, сельское хозяйство, Медицина, ф...
Name: profarea_names, dtype: object
```

```
In [20]: profareas = reduce(set.union, df.profarea_names, set())
```

```
In [21]: len(profareas)
```

```
Out[21]: 28
```

```
In [22]:
```

```
count_by_profarea = reduce(Counter.__add__, map(Counter, df.profarea_names))
```

```
In [23]: profareas_df = pd.DataFrame(count_by_profarea, index=['Количество вакансий']).T.reset_index()
```

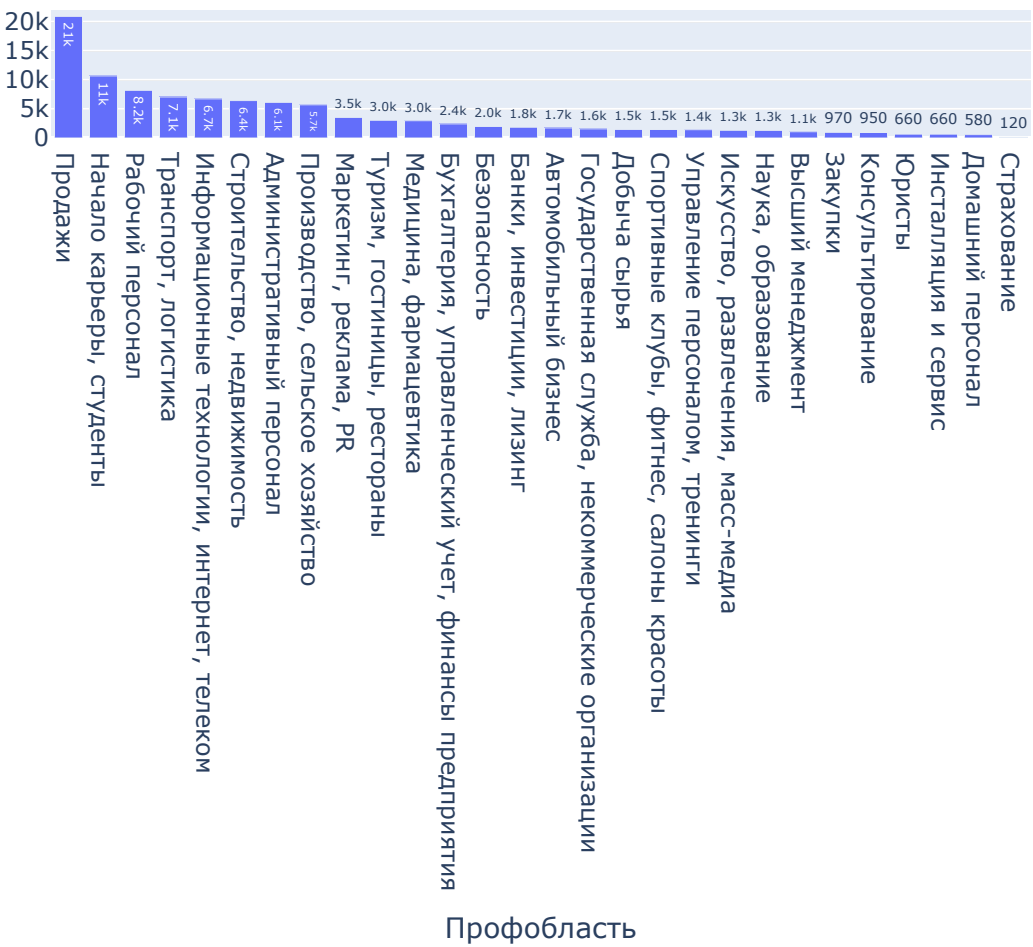
```
In [24]: ff.create_table(profareas_df)
```

Профобласть	Количество вакансий
Транспорт, логистика	7054
Административный персонал	6123
Домашний персонал	581
Спортивные клубы, фитнес, салоны красоты	1464
Бухгалтерия, управленческий учет, финансы предприятия	2427
Банки, инвестиции, лизинг	1846
Продажи	20903
Производство, сельское хозяйство	5710
Медицина, фармацевтика	2974
Начало карьеры, студенты	10674
Информационные технологии, интернет, телеком	6729
Маркетинг, реклама, PR	3538
Строительство, недвижимость	6432
Туризм, гостиницы, рестораны	3043
Наука, образование	1308
Рабочий персонал	8190
Искусство, развлечения, масс-медиа	1321
Закупки	971
Автомобильный бизнес	1733
Управление персоналом, тренинги	1431
Безопасность	1985
Консультирование	946
Страхование	120
Инсталляция и сервис	661
Юристы	663
Высший менеджмент	1080
Государственная служба, некоммерческие организации	1570
Добыча сырья	1473

In [25]:

```
px.bar(
    profareas_df,
    x='Профобласть',
    y='Количество вакансий',
    text_auto='.2s'
).update_xaxes(categoryorder='total descending')
```





Количество вакансий