

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

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

        import src.settings as settings
        from src.visualization.map import Point, Area
        from src.visualization.statistics import *
        from src.features.preprocessing import convert_salary
        from src.data.vacancy import Vacancy
```

```
In [2]: connect(
        host=settings.DB_HOST,
        port=settings.DB_PORT,
        db=settings.DB_NAME
    )
```

```
Out[2]: MongoClient(host=['localhost:27017'], document_class=dict, tz_aware=False, connect=True,
read_preference=Primary())
```

```
In [3]: pyo.init_notebook_mode()
```

```
In [4]: %load_ext autoreload
        %autoreload 2
```

## Получение данных

```
In [5]: 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',
                'specializations',
                'employer.name',
                'professional_roles',
                'key_skills',
            ])
    )
```

```
In [6]: df.set_index('_id', inplace=True)
```

## Обработка

### Удаление вакансий без зарплаты

```
In [7]: df['salary.to'].fillna(df['salary.from'], inplace=True)

df = df[df['salary.from'].notna()]
df = df[df['salary.to'].notna()]
df = df[df['salary.currency'].notna()]
```

```
In [8]: df['salary.currency'].isna().sum()
```

```
Out[8]: 0
```

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

```
Out[9]: (47440, 18)
```

### Перевод всех зарплат в рубли

```
In [10]: df[['salary.from', 'salary.to', 'salary.currency']] = df[['salary.from', 'salary.to', 'salary.currency']].apply(
    lambda row: [
        convert_salary(row['salary.from'], from_currency=row['salary.currency'], db=settings.DB),
        convert_salary(row['salary.to'], from_currency=row['salary.currency'], db=settings.DB),
        row['salary.currency']
    ], axis=1, result_type='expand')
```

```
In [11]: df['mean_salary'] = np.round((df['salary.to'] + df['salary.from']) / 2)
```

```
In [12]: df[df['salary.currency'] != 'RUR'].head(10)
```

Out[12]:

	description	key_skills	schedule.name	experience.id	experience.name	employment.name		
	_id							
	49810443	Требуемый опыт работы: 1–3 года Частичная заня...	[Internet, Голландский язык, Работа в команде,...	Удаленная работа	between1And3	От 1 года до 3 лет	Частичная занятость	8
	49226918	Сеть магазинов "Соседи" приглашает на работу з...	[Управление персоналом, Пользователь ПК, Работ...	Сменный график	between1And3	От 1 года до 3 лет	Полная занятость	3
	49225403	Обязанности: ручная бережная и качественная м...	[]	Полный день	noExperience	Нет опыта	Полная занятость	2
	49225631	Приглашаем водителей для работы в "Такси Алмаз...	[]	Гибкий график	noExperience	Нет опыта	Частичная занятость	4
	50155391	Обязанности: - Обработка входящего потока сооб...	[Грамотная речь, Пользователь ПК, Работа в ком...	Удаленная работа	noExperience	Нет опыта	Полная занятость	4
	50157216	Please note that by applying to this vacancy u...	[Английский язык, Coaching, Leadership Skills,...	Полный день	noExperience	Нет опыта	Стажировка	6
	50155707	Обязанности: Запуск рекламных компаний Google,...	[Английский язык, Маркетинговый анализ, Подгот...	Полный день	between1And3	От 1 года до 3 лет	Полная занятость	26
	50156232	Makeomatic расширяет свою команду! Уже 7 лет м...	[Git, JavaScript, Node.js, Docker, GitHub, Dev...	Полный день	between3And6	От 3 до 6 лет	Полная занятость	33
	50157118	Топ- менеджер/ Руководитель/ Управление (банковск...	[Работа в команде, CRM, Телефонные переговоры,...	Удаленная работа	between1And3	От 1 года до 3 лет	Полная занятость	33
	50157432	Julia Valler Event Staffing is a high-end mode...	[Английский язык, Работа в команде, MS PowerPo...	Полный день	noExperience	Нет опыта	Полная занятость	7

In [13]:

df.head(10)

Out[13]:

	description	key_skills	schedule.name	experience.id	experience.name	employment.nan
	_id					
49810439	Обязанности: Своевременная подача автомобиля; ...	[]	Полный день	between3And6	От 3 до 6 лет	Полная занятос
49810551	Обязанности: Уборка дома 500 кв.м., стирка, г...	[Русский язык, Чистоплотность]	Полный день	between1And3	От 1 года до 3 лет	Полная занятос
49810468	Студия Красоты и здоровья Кристалл ищет парикм...	[Пользователь ПК, Работа в команде, Грамотная ...	Полный день	between1And3	От 1 года до 3 лет	Полная занятос
45788942	Условия: ЗП от 50 тысяч на руки (оклад 22 тыся...	[Складская логистика, Терминалы Сбора Данных, ...	Сменный график	noExperience	Нет опыта	Полная занятос
49810601	Уважаемые соискатели, рассматриваются кандидат...	[]	Полный день	moreThan6	Более 6 лет	Полная занятос
49810507	Логопедический Пункт 1 приглашает Администрато...	[Обучение персонала, Пользователь ПК, Организа...	Полный день	between1And3	От 1 года до 3 лет	Полная занятос
49810469	Студия Красоты и здоровья Кристалл ищет парикм...	[Пользователь ПК, Работа в команде, Грамотная ...	Полный день	between1And3	От 1 года до 3 лет	Полная занятос
49810426	Обязанности: выполнение услуг массажа на высок...	[антицеллюлитный, класический, спортивный, лим...	Полный день	between3And6	От 3 до 6 лет	Полная занятос
47003369	Медиахолдинг "Май Медиа" ищет менеджера по про...	[Прямые продажи, Телефонные переговоры, Навыки...	Полный день	between1And3	От 1 года до 3 лет	Полная занятос
43592367	Обязанности: Запрос цен и анализ по счетам от...	[MS PowerPoint, MS Access, Работа с базами дан...	Сменный график	noExperience	Нет опыта	Полная занятос

Добавление средней зарплаты

In [14]:

```
total_salary = df['mean_salary'].sum()
total_salary
```

Out[14]:

2782463002.0

In [15]:

```
total_salary_by_area = df[['area.name', 'mean_salary']].groupby(['area.name'], as_index=False,
    columns={'mean_salary': 'total_salary'})
total_salary_by_area.head(10)
```

Out[15]:

	area.name	total_salary
0	Абаза	770000.0
1	Абай	86000.0
2	Абакан	8340594.0
3	Абан	119921.0
4	Абатское	195048.0
5	Абинск	65000.0
6	Авсюнино	131500.0
7	Агалатово	30000.0
8	Агаповка	86000.0
9	Агеево	66000.0

# Анализ

## Разделение зарплат по городам

In [16]:

```
other = total_salary_by_area.total_salary < (total_salary / 100) # общая зарплата меня
other_value = total_salary_by_area.total_salary[other].agg('sum')
total_salary_by_area = total_salary_by_area[~other]
total_salary_by_area = total_salary_by_area.append({'area.name': 'Другие регионы', 'total_salary': other_value, 'ignore_index=True})
```

In [17]:

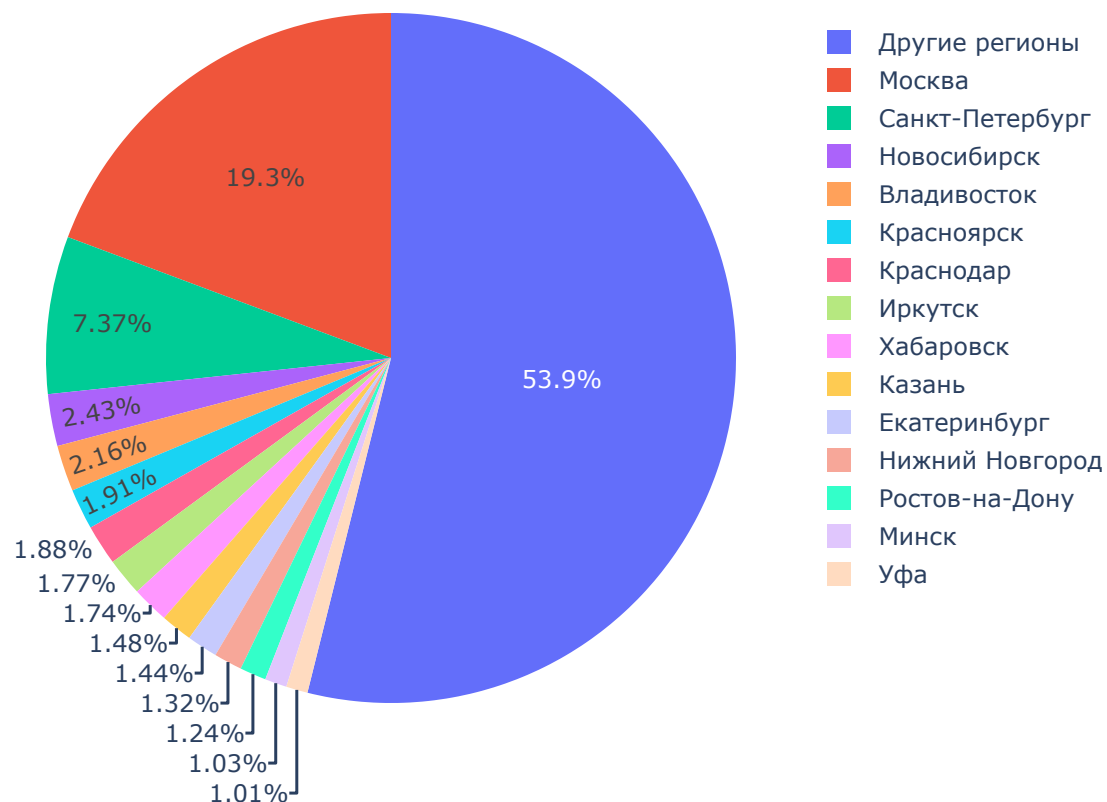
```
total_salary_by_area
```

Out[17]:

	area.name	total_salary
0	Владивосток	6.015154e+07
1	Екатеринбург	4.013486e+07
2	Иркутск	4.925897e+07
3	Казань	4.131740e+07
4	Краснодар	5.243045e+07
5	Красноярск	5.313064e+07
6	Минск	2.859788e+07
7	Москва	5.370685e+08
8	Нижний Новгород	3.684292e+07
9	Новосибирск	6.767413e+07
10	Ростов-на-Дону	3.439070e+07
11	Санкт-Петербург	2.051561e+08
12	Уфа	2.819120e+07
13	Хабаровск	4.851375e+07
14	Другие регионы	1.499604e+09

```
In [18]: px.pie(total_salary_by_area, names='area.name', values='total_salary', title='Разделение
```

## Разделение зарплат по городам

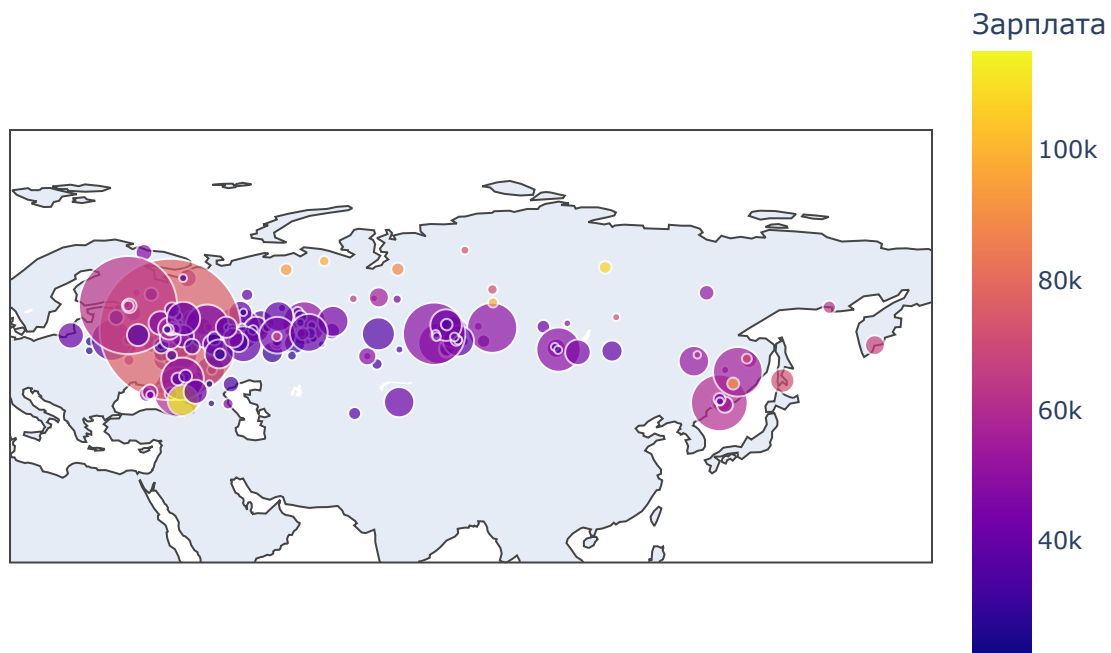


## Количество вакансий и средняя зарплата относительно местоположения

```
In [19]: geo_df = df.reset_index()[['_id', 'address.city', 'address.lat', 'address.lng', 'mean_salary'])
geo_df.groupby('address.city', as_index=False)\
    .agg({'mean_salary': 'mean', '_id': 'count', 'address.lat': 'mean', 'address.lng': 'mean'})\
    .rename(columns={'_id': 'count'})
```

```
In [20]: px.scatter_geo(
    geo_df[(geo_df['count'] > 10) & (geo_df['mean_salary'] < 200_000)],
    lat='address.lat',
    lon='address.lng',
    size='count',
    fitbounds='locations',
    color='mean_salary',
    hover_data=['address.city'],
    center={'lat': 53, 'lon': 83},
    size_max=50,
    labels={'mean_salary': 'Зарплата'},
    title='Количество вакансий и средняя зарплата относительно города'
)
```

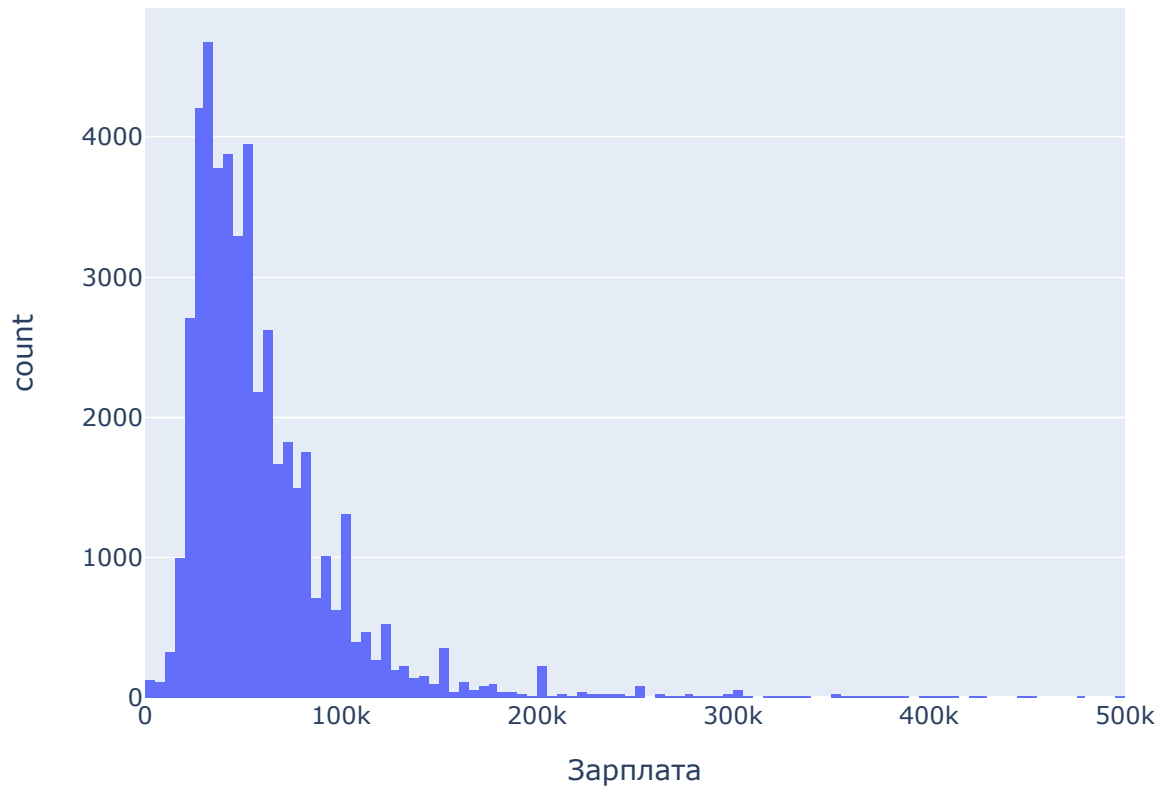
## Количество вакансий и средняя зарплата относительно города



## Распределения

```
In [21]: px.histogram(  
    df[df.mean_salary < 500_000],  
    x='mean_salary',  
    nbins=100,  
    title='Распределение зарплат',  
    labels={'mean_salary': 'Зарплата'})
```

## Распределение зарплат

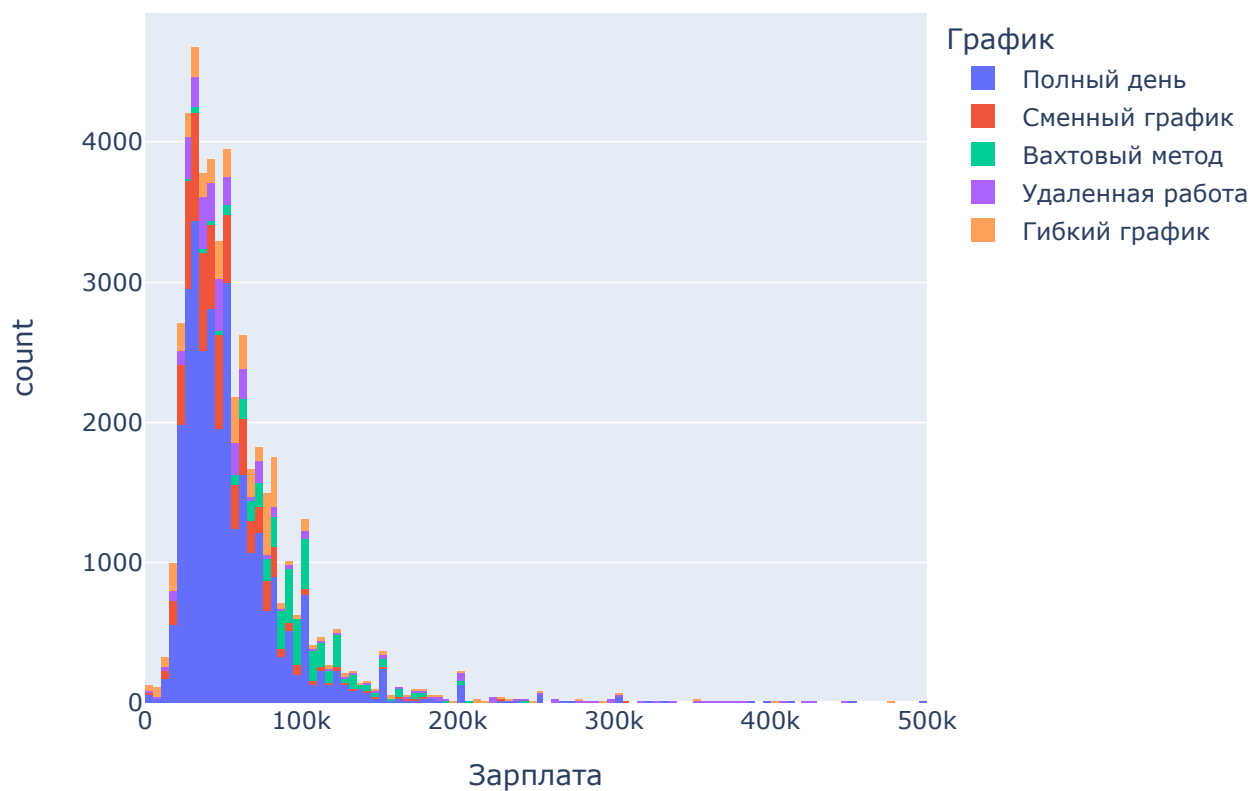


In [22]:

```
px.histogram(  
    df[df.mean_salary < 500_000],  
    x='mean_salary',  
    color='schedule.name',  
    nbins=100,  
    title='Распределение зарплат с учетом графика работы',  
    labels={'mean_salary': 'Зарплата', 'schedule.name': 'График'})
```



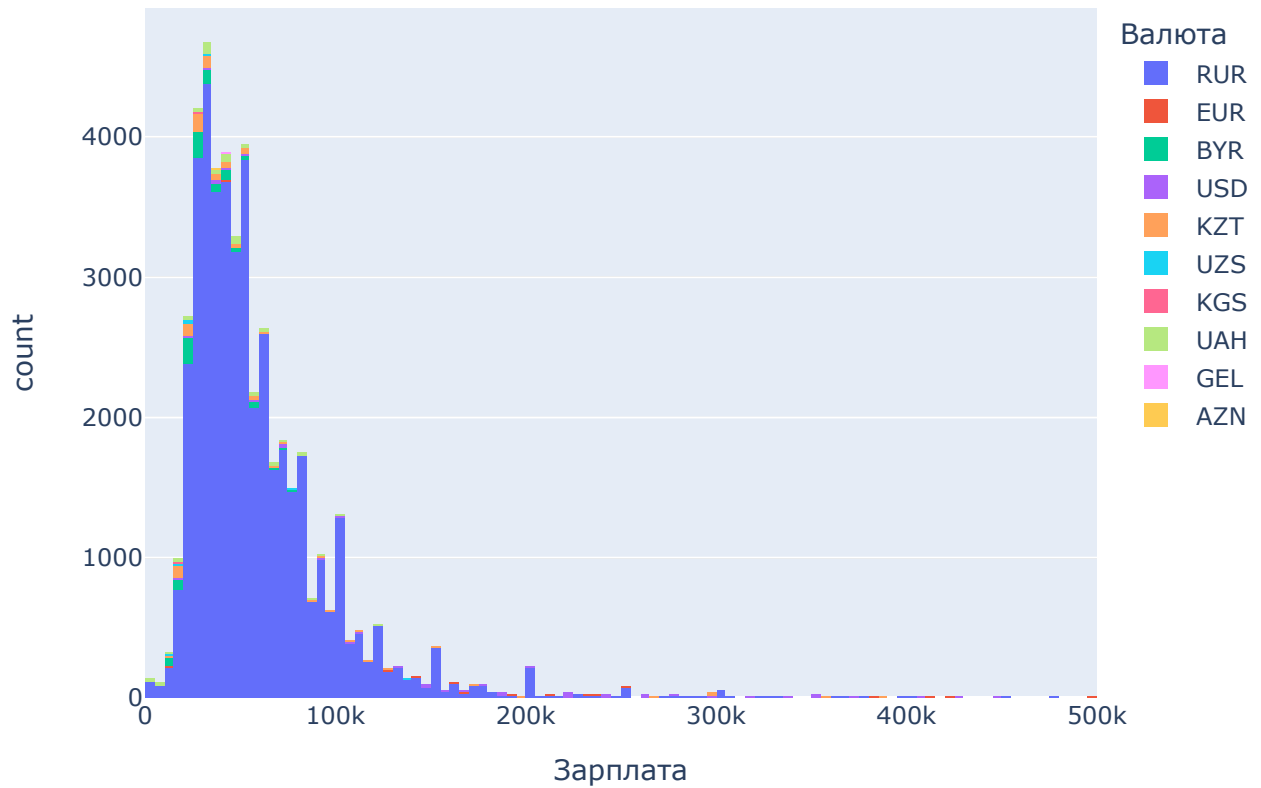
## Распределение зарплат с учетом графика работы



In [23]:

```
px.histogram(  
    df[df.mean_salary < 500_000],  
    x='mean_salary',  
    color='salary.currency',  
    nbins=100,  
    title='Распределение зарплат с учетом валюты работы',  
    labels={'mean_salary': 'Зарплата', 'salary.currency': 'Валюта'})
```

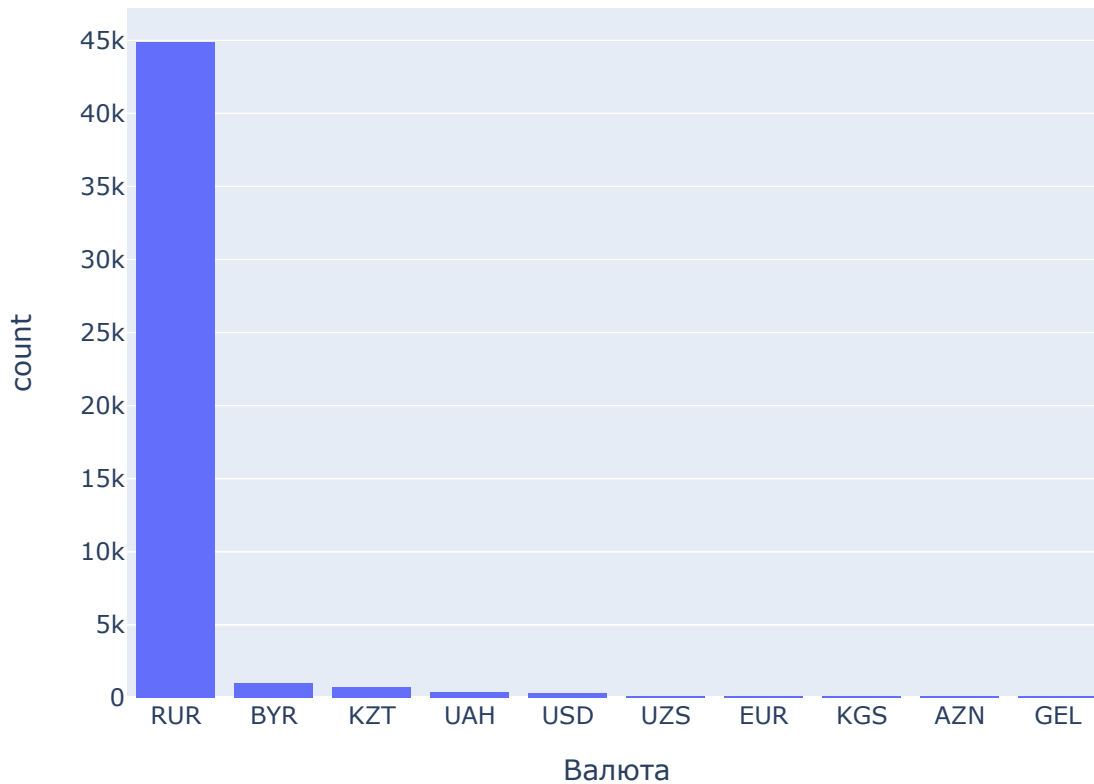
## Распределение зарплат с учетом валюты работы



In [24]:

```
px.histogram(  
    df,  
    x='salary.currency',  
    title='Количество вакансий для каждой валюты',  
    labels={'salary.currency': 'Валюта'})  
) .update_xaxes(categoryorder='total descending')
```

## Количество вакансий для каждой валюты



## Анализ зарплат относительно навыков

```
In [25]: key_skills = reduce(set.union, df.key_skills, set())
```

```
In [26]: len(key_skills)
```

```
Out[26]: 7364
```

```
In [27]: key_skill_df = pd.DataFrame((
    {
        'Навык': key_skill,
        '**df[df['key_skills'].map({key_skill}.issubset)][mean_salary']
        .agg(['count', 'sum', 'mean', 'max', 'min']).to_dict()
    }
    for key_skill in key_skills
))
```

```
In [28]: ff.create_table(key_skill_df[key_skill_df['Навык'].map(len) < 30].head(50).astype(int, c
```

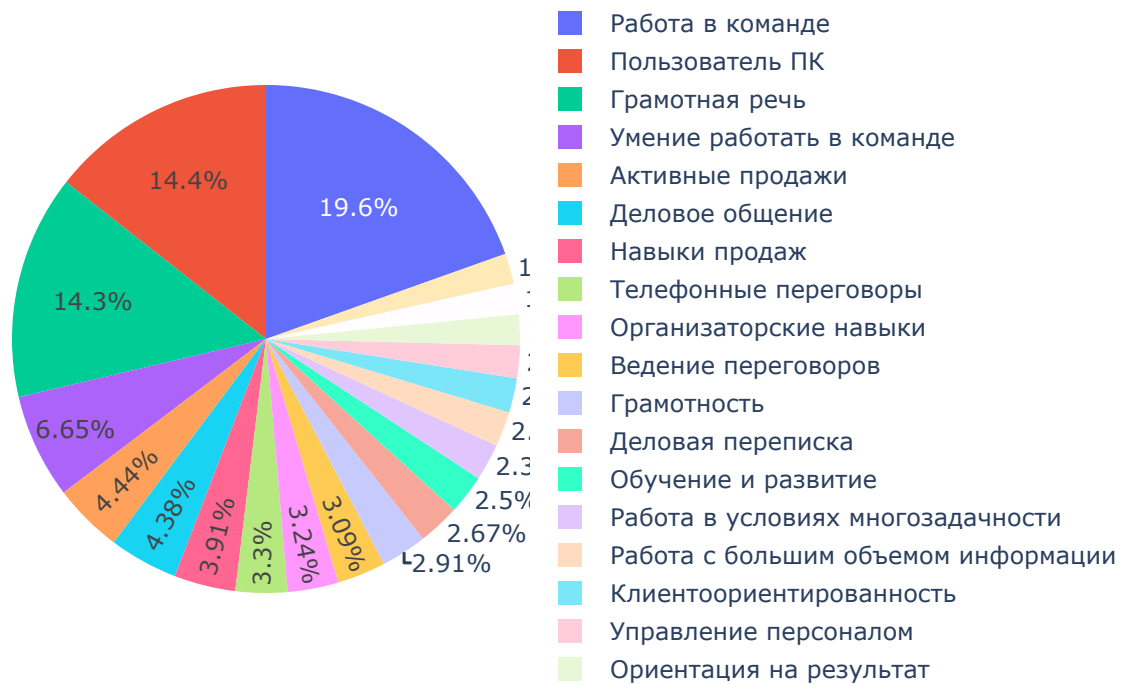
Навык	count	sum	mean	max	min
инструменты моделирования	4	40000	40000	40000	40000
Поиск объектов	20	2025674	101283	250000	30000
#Специалист_Call-центра	1	53500	53500	53500	53500
Нефтегаз	2	180000	90000	90000	90000
Лазерная техника	8	478806	59850	170000	24680
Учет остатков	114	5917456	51907	167500	22500
Автоэлектрик	3	242000	80666	123500	57500
реклама на радио	1	96000	96000	96000	96000
Взятие крови	7	468500	66928	92500	26000
выкуп автомобилей	2	220000	110000	120000	100000
Штрихкодирование	3	112500	37500	40000	35000
шелкограф	1	62500	62500	62500	62500
новостройки	3	295000	98333	110000	75000
Вывод компании из рынка	16	1722500	107656	150000	30000
Водитель на личном автомобиле	7	266000	38000	38000	38000
Электронные торговые площадки	1	70000	70000	70000	70000
Умение обучать других людей	4	244000	61000	100000	48000
Умение работать на результат	4	188636	47159	60000	27500
обслуживание карьеров	1	90000	90000	90000	90000
Звукорежиссура	1	22500	22500	22500	22500
Multithread Programming	1	300000	300000	300000	300000
CPC	2	135000	67500	105000	30000
Нотариус	1	80000	80000	80000	80000
Разгрузка	1	29035	29035	29035	29035
NX	1	30000	30000	30000	30000
Арбитраж трафика	1	180000	180000	180000	180000
B2B маркетинг	39	3797688	97376	250000	20000
kfc	16	560000	35000	35000	35000
Навыки презентации	250	21394920	84564	500000	140
Безопасность	64	3843450	60053	120000	2200
валютные счета	1	55000	55000	55000	55000
Blender 3D	7	610000	87142	112500	55000
Навыки мотивации персонала	2	75000	37500	40000	35000

Костюмы	1	80000	80000	80000	80000
Агрохимия	4	215000	53750	70000	45000
Анатомия	3	265000	88333	120000	70000
Судостроение	1	120000	120000	120000	120000
Гостиничный менеджмент	8	414444	51805	70000	34944
Усольский химфарм завод	2	69160	34580	34580	34580
ворота	1	110000	110000	110000	110000
добропорядочность	1	25000	25000	25000	25000
Бодибилдинг	1	120000	120000	120000	120000
Знание 230-ФЗ	1	35000	35000	35000	35000
Data science	1	145000	145000	145000	145000
разработка эскизов	1	42842	42842	42842	42842
оператор ПУ	1	134800	134800	134800	134800
Media Buying	2	480000	240000	300000	180000
Бумажный документооборот	41	1676958	40901	72000	13500
холодильное оборудование	1	28000	28000	28000	28000
GPS	1	90000	90000	90000	90000

In [29]:

```
px.pie(
    key_skill_df[key_skill_df['count'] > 1_000],
    names='Навык',
    values='count',
    title='Доля вакансий для самых популярных навыков',
    labels={'count': 'Количество вакансий'})
```

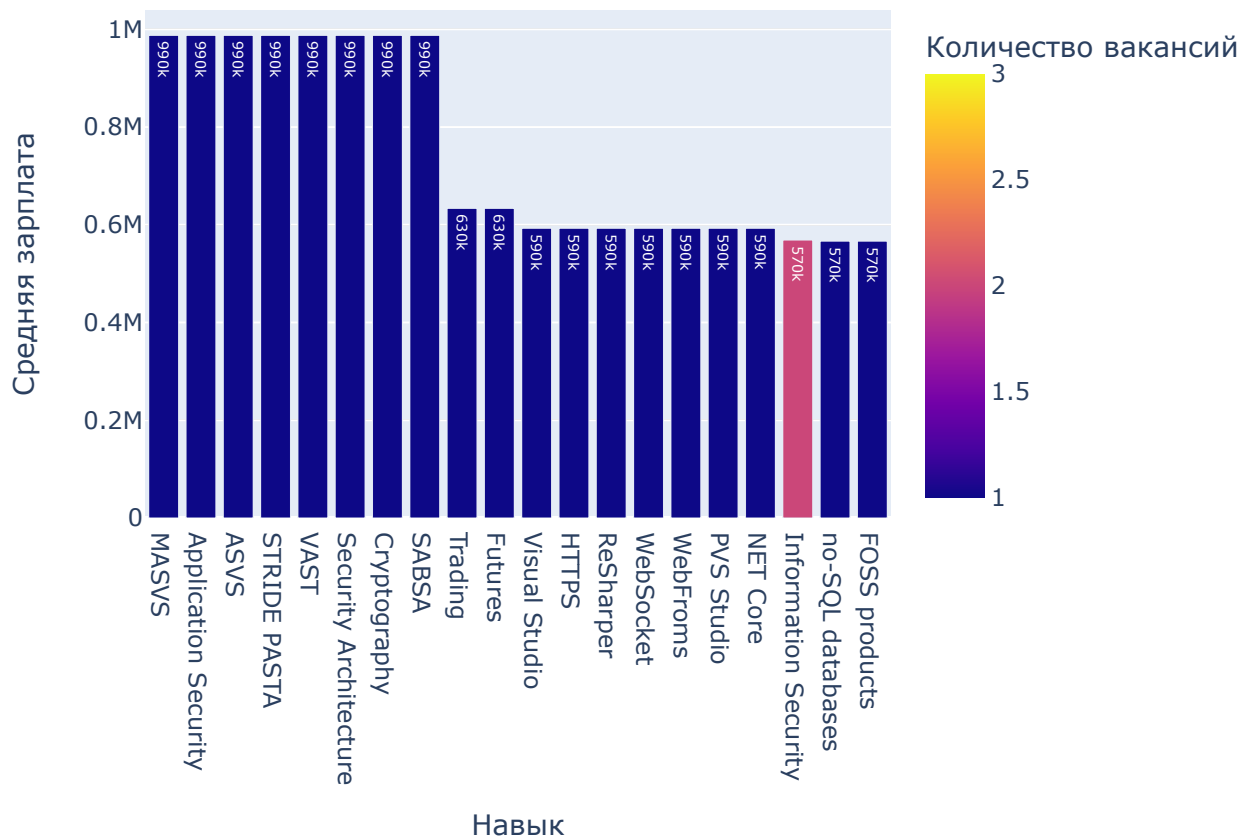
## Доля вакансий для самых популярных навыков



```
In [30]: sorted_key_skill_df = key_skill_df.sort_values('mean', ascending=False)
```

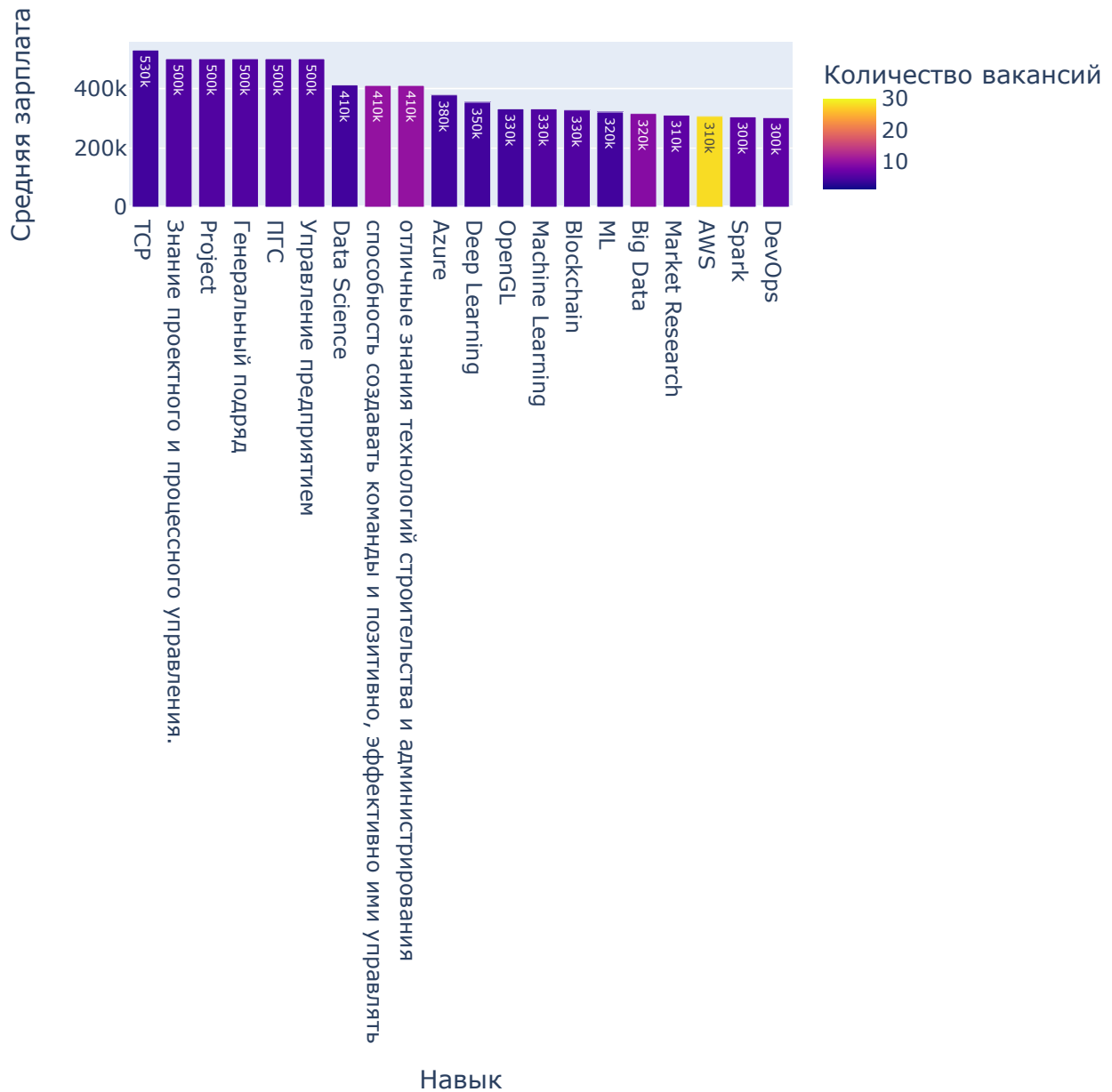
```
In [31]: px.bar(
    sorted_key_skill_df.head(20),
    x='Навык',
    y='mean',
    color='count',
    title='Средняя зарплата самых высокооплачиваемых навыков',
    labels={'mean': 'Средняя зарплата', 'count': 'Количество вакансий'},
    range_color=(1, 3),
    text_auto='.2s'
).update_xaxes(categoryorder='total descending')
```

## Средняя зарплата самых высокооплачиваемых навыков



```
In [32]: px.bar(
    sorted_key_skill_df[sorted_key_skill_df['count'] >= df.shape[0] * 0.008 * 0.01].head(
    x='Навык',
    y='mean',
    color='count',
    title='Средняя зарплата самых высокооплачиваемых навыков, необходимых не менее чем 1',
    labels={'mean': 'Средняя зарплата', 'count': 'Количество вакансий'},
    range_color=(1, 30),
    text_auto='.2s',
    height=700,
).update_xaxes(categoryorder='total descending')
```

## Средняя зарплата самых высокооплачиваемых навыков, необходимых

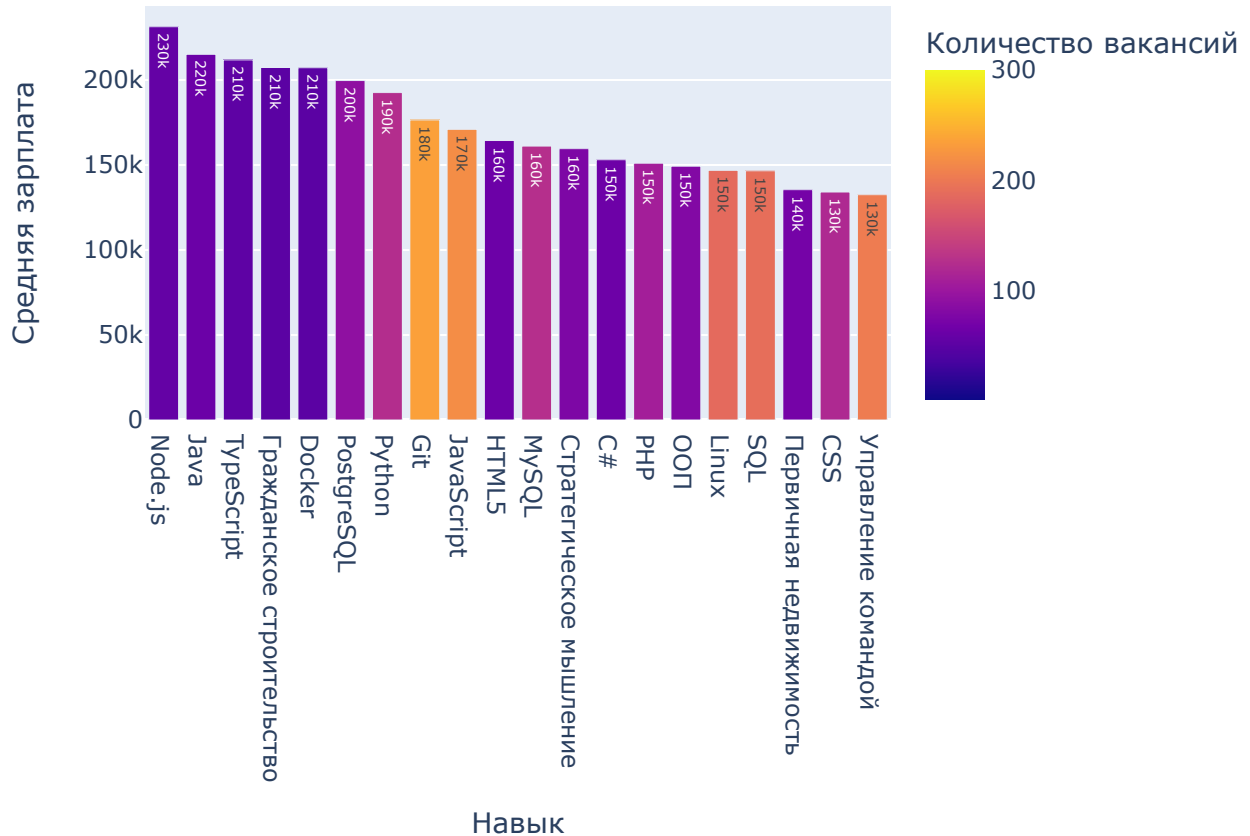


In [33]:

```
px.bar(
    sorted_key_skill_df[sorted_key_skill_df['count'] >= df.shape[0] * 0.1 * 0.01].head(20),
    x='Навык',
    y='mean',
    color='count',
    title='Средняя зарплата самых высокооплачиваемых навыков, необходимых не менее чем 10 вакансиям',
    labels={'mean': 'Средняя зарплата', 'count': 'Количество вакансий'},
    range_color=(1, 300),
    text_auto='.2s'
).update_xaxes(categoryorder='total descending')
```



## Средняя зарплата самых высокооплачиваемых навыков, необходимых



## Анализ зарплат относительно профобластей

```
In [34]: specs_df = df.copy()

specs_df.specializations = specs_df.specializations.map(lambda specs: list(map(itemgetter(0), specs)))
specs_df = specs_df[specs_df.specializations.notna()]

specs_df['specialization_profarea_names'] = df.specializations.map(
    lambda specs: list(set(map(itemgetter('profarea_name'), specs))))
specs_df = specs_df[specs_df.specialization_profarea_names.notna()]

specs_df['full_specializations'] = df.specializations
```

```
In [35]: specs_df[['specialization_profarea_names', 'specializations']].head(10)
```

Out[35]:

	specialization_profarea_names	specializations
_id		
49810439	[Транспорт, логистика]	[Автоперевозки, Водитель, Логистика, Экспедитор]
49810551	[Административный персонал, Домашний персонал]	[Уборщица/уборщик, домработница/домработник, Г...
49810468	[Спортивные клубы, фитнес, салоны красоты]	[Парикмахер]
45788942	[Транспорт, логистика]	[Кладовщик, Рабочий склада, Логистика]
49810601	[Банки, инвестиции, лизинг, Бухгалтерия, управ...	[Учет заработной платы, Основные средства, Нал...
49810507	[Административный персонал]	[Управляющий офисом (Office manager), Персонал...
49810469	[Спортивные клубы, фитнес, салоны красоты]	[Парикмахер]
49810426	[Спортивные клубы, фитнес, салоны красоты]	[Массажист]
47003369	[Продажи]	[Прямые продажи, Менеджер по работе с клиентами]
43592367	[Бухгалтерия, управленческий учет, финансы пре...	[Аудит, Другое, Финансовый анализ]

In [36]:

```
all_specialization_names = list(reduce(set.union, specs_df.specializations, set()))
all_specialization_names[:15]
```

Out[36]:

```
['Учет заработной платы',
 'Системная интеграция',
 'Шлифовщик',
 'Основные средства',
 'Корпоративное право',
 'Охранник',
 'Ценные бумаги',
 'Авторское право',
 'Журналистика',
 'Фармацевтическая промышленность',
 'Фотография',
 'Интернет',
 'Кассир, Инкассатор',
 'Отопление, вентиляция и кондиционирование',
 'Таможенное оформление']
```

In [37]:

```
len(all_specialization_names)
```

Out[37]:

```
504
```

In [38]:

```
profarea_by_specialization = reduce(
    dict.__or__,
    map(
        lambda specs:
            dict(zip(
                map(itemgetter('name'), specs),
                map(itemgetter('profarea_name'), specs)
            )),
        specs_df.full_specializations
    )
)
```

In [39]:

```
ff.create_table(['Специализация', 'Профобласть']) + list(profarea_by_specialization.it
```

Специализация	Профобласть	
Автоперевозки	Транспорт, логистика	
Водитель	Транспорт, логистика	
Логистика	Транспорт, логистика	
Экспедитор	Транспорт, логистика	
Уборщица/уборщик	Административный персонал	
домработница/домработник, Горничная	Домашний персонал	
Парикмахер	Спортивные клубы, фитнес, салоны красоты	
Кладовщик	Транспорт, логистика	
Рабочий склада	Транспорт, логистика	
Учет заработной платы	Бухгалтерия, управленческий учет, финансы предприятия	
Основные средства	Бухгалтерия, управленческий учет, финансы предприятия	
Налоги	Бухгалтерия, управленческий учет, финансы предприятия	
Руководство бухгалтерией	Бухгалтерия, управленческий учет, финансы предприятия	
Расчет себестоимости	Бухгалтерия, управленческий учет, финансы предприятия	
Управляющий офисом (Office manager)	Административный персонал	
Персональный ассистент	Административный персонал	
Делопроизводство	Административный персонал	
Ресепшен	Административный персонал	
Вечерний секретарь	Административный персонал	
Массажист	Спортивные клубы, фитнес, салоны красоты	
Прямые продажи	Продажи	
Менеджер по работе с клиентами	Продажи	
Аудит	Бухгалтерия, управленческий учет, финансы предприятия	
Другое	Рабочий персонал	
Финансовый анализ	Бухгалтерия, управленческий учет, финансы предприятия	
Розничная торговля	Продажи	
Дистрибуция	Продажи	
Оптовая торговля	Продажи	
Продукты питания	Продажи	
Алкоголь	Продажи	
Мерчендайзинг	Маркетинг, реклама, PR	
Агент	Строительство, недвижимость	
Персонал кухни	Туризм, гостиницы, рестораны	

Воспитатель, Гувернантка/гувернёр, Няня	Домашний персонал	
Преподавание	Наука, образование	
Продажи	Начало карьеры, студенты	
Продажи по телефону, Телемаркетинг	Продажи	
Сантехник	Рабочий персонал	
Начальный уровень, Мало опыта	Производство, сельское хозяйство	
Секретарь	Административный персонал	
Грузчик	Рабочий персонал	
Комплектовщик, Укладчик-упаковщик	Рабочий персонал	
Складское хозяйство	Транспорт, логистика	
ГСМ, нефть, бензин	Продажи	
Гардеробщик	Рабочий персонал	
Дворник, Уборщик	Рабочий персонал	
Ввод данных	Административный персонал	
Сотрудник call-центра	Административный персонал	
Услуги для бизнеса	Продажи	
Эксплуатация	Строительство, недвижимость	

In [40]:

count\_by\_specialization = {spec: specs\_df.specializations.map({spec}.issubset).sum() for spec in specs\_df.specializations}
count\_by\_specialization = Counter(count\_by\_specialization)

In [41]:

ff.create\_table(['Специализация', 'Количество вакансий']) + count\_by\_specialization.most\_common(10)

Специализация	Количество вакансий
Розничная торговля	10704
Начальный уровень, Мало опыта	9135
Торговые сети	8017
Продукты питания	7397
Продажи	6681
Продавец в магазине	6055
Менеджер по работе с клиентами	5314
Прямые продажи	4366
Другое	3127
Водитель	2872

In [42]:

```
spec_names_df = pd.DataFrame((
    {
        'Специализация': spec,
        'Профобласть': profarea_by_specialization[spec],
        **specs_df[specs_df.specializations.map({spec}.issubset)][['mean_salary']]
            .agg(['count', 'sum', 'mean', 'max', 'min']).to_dict()
    }
    for spec in all_specialization_names
))
```

In [43]:

```
ff.create_table(spec_names_df[(spec_names_df['Специализация'].map(len) < 20) & (spec_nar
```

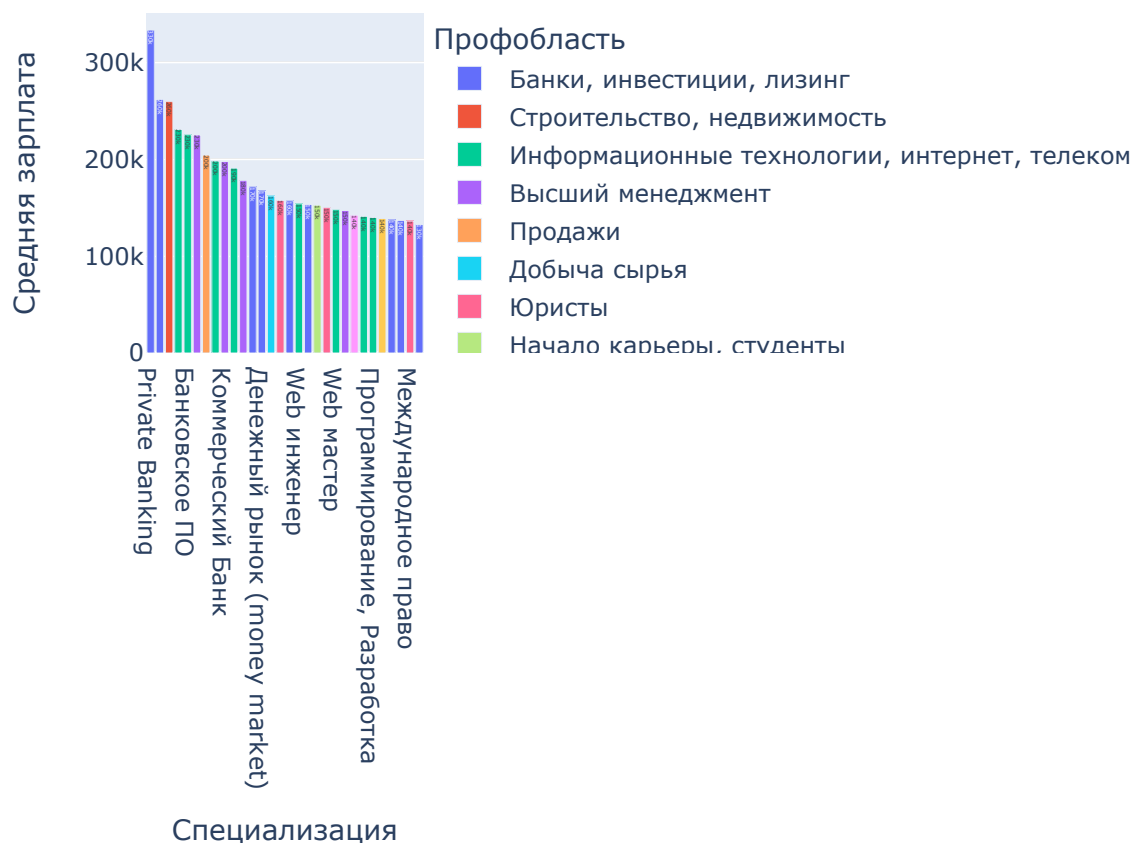
Специализация	Профобласть	count	sum	mean	max	min
Шлифовщик	Рабочий персонал	151.0	846742.0	56449.46666666666	60000.0	25000.0
Корпоративное право	Юристы	214.0	11801490.0	55147.14953271008	740000.0	40.0
Охранник	Безопасность	1184.0	62719487.0	52972.53969594592	40500.0	1500.0
Авторское право	Юристы	2.0	315000.0	157500.0	225000.0	90000.0
Юриспруденция	Высший менеджмент	5.0	302560.0	60512.0	73500.0	40000.0
Оператор станков	Рабочий персонал	484.0	29126539.0	58960.60526315789	30000.0	35.0
Трудовое право	Юристы	217.0	11705390.0	53941.88940092066	740000.0	40.0
Договорное право	Юристы	234.0	12966390.0	55411.92307692308	740000.0	40.0
Кузнец	Рабочий персонал	31.0	183000.0	61000.0	95000.0	40000.0
Уголь	Добыча сырья	180.0	14820560.0	82336.44444444444	415000.0	19000.0
Продавец в магазине	Продажи	6055.0	208877200.0	34496.64739884259	42500.0	35.0
Оптовая торговля	Продажи	2395.0	151670552.0	63327.99665974570	740000.0	140.0
Нефть	Добыча сырья	446.0	42257440.0	94747.62331839665	395000.0	20500.0
Юрисконсульт	Юристы	283.0	15135890.0	53483.71024734980	740000.0	40.0
Проводник	Рабочий персонал	21.0	49000.0	24500.0	27000.0	22000.0
Инвестиции	Высший менеджмент	2.0	450000.0	225000.0	250000.0	200000.0
Дистрибуция	Продажи	442.0	27018712.0	61128.30769230769	307000.0	1100.0
Токарь, Фрезер	Рабочий персонал	362.0	21840626.0	72319.9536423500	300000.0	20000.0
Банковское право	Юристы	9.0	544604.0	60511.55555555555	525000.0	30000.0
Клининговые услуги	Продажи	16.0	675670.0	42229.375	135000.0	2000.0
Прямые продажи	Продажи	4366.0	287423601.0	65832.24942739825	392500.0	45.0
Торговые сети	Продажи	8017.0	300091246.0	37431.86304158709	587000.0	35.0
Мебель	Продажи	220.0	14673553.0	66697.96818181818	812500.0	40.0
Механик	Рабочий персонал	722.0	43656866.0	60466.57340720850	205000.0	40.0
Грузчик	Рабочий персонал	1392.0	75372097.0	54146.62140802527	205000.0	1402.0
Слесарь	Рабочий персонал	814.0	44995461.0	55276.97911547850	178000.0	40.0
Коммерческий банк	Высший менеджмент	27.0	5335897.0	197625.8148148000	480000.0	66000.0
Цветные металлы	Продажи	15.0	1150536.0	76702.4	150000.0	30000.0
Страховое право	Юристы	2.0	147300.0	73650.0	92300.0	55000.0
Столяр	Рабочий персонал	88.0	4572084.0	57151.05	100000.0	16000.0
Алкоголь	Продажи	157.0	6296857.0	40107.36942675059	750000.0	10000.0
Физика	Наука, образование	15.0	718278.0	47885.2	90000.0	15000.0
Сварщик	Рабочий персонал	581.0	59455657.0	102333.31669282000	525000.0	24848.0

Продукты питания	Продажи	7397.0	295798132.0	39988.932269858000.0	58000.0	1100.0
Франчайзинг	Продажи	4.0	392500.0	98125.0	160000.0	32500.0
Геологоразведка	Добыча сырья	91.0	10522762.0	115634.74725274000.0	273000.0	25000.0
Андеррайтер	Страхование	1.0	60000.0	60000.0	60000.0	60000.0
Инженерные науки	Наука, образование	61.0	2726874.0	44702.8524590116000.0	116000.0	15000.0
Земельное право	Юристы	14.0	732000.0	52285.71428571428.0	75000.0	25000.0
ГСМ, нефть, бензин	Продажи	97.0	5812067.0	59918.216494815500.0	815500.0	8575.0
Гардеробщик	Рабочий персонал	10.0	248850.0	24885.0	37500.0	14000.0
Ювелир	Рабочий персонал	10.0	585240.0	58524.0	85000.0	17240.0
Администрирование	Высший менеджмент	61.0	5585810.0	87278.28125	300000.0	20000.0
Гуманитарные науки	Наука, образование	299.0	18743223.0	46975.496240600000.0	600000.0	51.0
Семейное право	Юристы	15.0	619500.0	41300.0	110000.0	22500.0
Compliance	Юристы	2.0	142500.0	71250.0	100000.0	42500.0
Международное право	Юристы	4.0	545000.0	136250.0	350000.0	40000.0
Преподавание	Наука, образование	605.0	27518091.0	45484.447933800000.0	380000.0	51.0
Разнорабочий	Рабочий персонал	1170.0	83441422.0	71317.4547008282000.0	850000.0	1402.0
Металлопрокат	Продажи	42.0	3389589.0	80704.5	200000.0	35000.0

In [44]:

```
px.bar(
    spec_names_df.sort_values('mean', ascending=False).head(30),
    x='Специализация',
    y='mean',
    color='Профобласть',
    labels={'mean': 'Средняя зарплата'},
    title='30 самых высокооплачиваемых специализаций',
    text_auto='.2s'
).update_xaxes(categoryorder='total descending')
```

## 30 самых высокооплачиваемых специализаций



```
In [45]: geo_profareas_df = pd.concat(map(
    pd.DataFrame,
    (dict(zip(
        ('profarea', 'lat', 'lng'),
        zip(*zip(
            spec_names := set(map(itemgetter('profarea_name'), specs)),
            [lat] * len(spec_names),
            [lng] * len(spec_names)
        ))
    ))
    for lat, lng, specs in df[['address.lat', 'address.lng', 'specializations']].dropna
)).reset_index().drop('index', axis=1)
```

```
In [46]: ff.create_table(geo_profareas_df.head(15))
```



profarea	lat	lng
Административный персонал	59.932428	30.439198
Домашний персонал	59.932428	30.439198
Транспорт, логистика	55.52063	37.774149
Административный персонал	48.745092	44.499916
Спортивные клубы, фитнес, салоны красоты	55.718796	37.598825
Продажи	57.001064	40.968217
Бухгалтерия, управленческий учет, финансы предприятия	50.576507	36.578904
Строительство, недвижимость	54.169395	37.58279
Наука, образование	55.789633	49.134055
Домашний персонал	55.789633	49.134055
Продажи	59.947991	30.268051
Информационные технологии, интернет, телеком	59.947991	30.268051
Транспорт, логистика	56.358208	37.505678
Рабочий персонал	56.358208	37.505678
Административный персонал	56.836101	60.614578

In [47]:

```
# noinspection PyTypeChecker
points = reduce(list.__add__, [
    list(map(Point.from_tuple, zip(
        spec_names := set(map(itemgetter('profarea_name'), specs)),
        [lat] * len(spec_names),
        [lng] * len(spec_names)
    )))
    for lat, lng, specs in df[['address.lat', 'address.lng', 'specializations']].dropna
])
```

In [48]:

```
points[:100]
```

```
Out[48]: Point(profarea='Административный персонал', lat=59.932428, lng=30.439198),
Point(profarea='Домашний персонал', lat=59.932428, lng=30.439198),
Point(profarea='Транспорт, логистика', lat=55.52063, lng=37.774149),
Point(profarea='Административный персонал', lat=48.745092, lng=44.499916),
Point(profarea='Спортивные клубы, фитнес, салоны красоты', lat=55.778796, lng=37.59882
5),
Point(profarea='Продажи', lat=57.001064, lng=40.968217),
Point(profarea='Бухгалтерия, управленческий учет, финансы предприятия', lat=50.576507,
lng=36.578904),
Point(profarea='Строительство, недвижимость', lat=54.169395, lng=37.58279),
Point(profarea='Наука, образование', lat=55.789633, lng=49.134055),
Point(profarea='Домашний персонал', lat=55.789633, lng=49.134055),
Point(profarea='Продажи', lat=59.947991, lng=30.268051),
Point(profarea='Информационные технологии, интернет, телеком', lat=59.947991, lng=30.26
8051),
Point(profarea='Транспорт, логистика', lat=56.358208, lng=37.505678),
Point(profarea='Рабочий персонал', lat=56.358208, lng=37.505678),
Point(profarea='Административный персонал', lat=56.836101, lng=60.614578),
Point(profarea='Продажи', lat=56.836101, lng=60.614578),
Point(profarea='Продажи', lat=55.689695, lng=37.467482),
Point(profarea='Продажи', lat=55.8623392366, lng=37.5676616141),
Point(profarea='Строительство, недвижимость', lat=44.740913, lng=37.724526),
Point(profarea='Медицина, фармацевтика', lat=53.499701, lng=49.275127),
Point(profarea='Продажи', lat=44.740913, lng=37.724526),
Point(profarea='Административный персонал', lat=55.558208, lng=37.695501),
Point(profarea='Продажи', lat=55.558208, lng=37.695501),
Point(profarea='Спортивные клубы, фитнес, салоны красоты', lat=55.558208, lng=37.69550
1),
Point(profarea='Искусство, развлечения, масс-медиа', lat=59.9410372053, lng=30.27677245
27),
Point(profarea='Маркетинг, реклама, PR', lat=59.9410372053, lng=30.2767724527),
Point(profarea='Туризм, гостиницы, рестораны', lat=59.935638, lng=30.361673),
Point(profarea='Продажи', lat=55.039068, lng=82.971696),
Point(profarea='Продажи', lat=55.890239, lng=37.422548),
Point(profarea='Административный персонал', lat=54.903112, lng=52.304937),
Point(profarea='Продажи', lat=54.903112, lng=52.304937),
Point(profarea='Спортивные клубы, фитнес, салоны красоты', lat=54.903112, lng=52.30493
7),
Point(profarea='Медицина, фармацевтика', lat=55.439242, lng=37.56635),
Point(profarea='Управление персоналом, тренинги', lat=53.184521, lng=50.104092),
Point(profarea='Медицина, фармацевтика', lat=55.439242, lng=37.56635),
Point(profarea='Транспорт, логистика', lat=55.750358, lng=37.755895),
Point(profarea='Строительство, недвижимость', lat=51.686534, lng=39.256746),
Point(profarea='Продажи', lat=51.686534, lng=39.256746),
Point(profarea='Спортивные клубы, фитнес, салоны красоты', lat=55.856341, lng=37.49384
7),
Point(profarea='Туризм, гостиницы, рестораны', lat=59.928438, lng=30.35887),
Point(profarea='Административный персонал', lat=59.928438, lng=30.35887),
Point(profarea='Туризм, гостиницы, рестораны', lat=59.7999429934, lng=30.271653709),
Point(profarea='Автомобильный бизнес', lat=56.129504, lng=47.300037),
Point(profarea='Медицина, фармацевтика', lat=54.715909, lng=20.484795),
Point(profarea='Начало карьеры, студенты', lat=53.899117, lng=27.524919),
Point(profarea='Автомобильный бизнес', lat=53.899117, lng=27.524919),
Point(profarea='Транспорт, логистика', lat=55.825391, lng=37.353971),
Point(profarea='Начало карьеры, студенты', lat=47.21933, lng=39.713827),
Point(profarea='Продажи', lat=47.21933, lng=39.713827),
Point(profarea='Строительство, недвижимость', lat=45.014256, lng=38.956242),
Point(profarea='Производство, сельское хозяйство', lat=45.014256, lng=38.956242),
Point(profarea='Продажи', lat=55.531188, lng=37.620465),
Point(profarea='Инсталляция и сервис', lat=56.079904, lng=93.011349),
Point(profarea='Производство, сельское хозяйство', lat=56.079904, lng=93.011349),
Point(profarea='Спортивные клубы, фитнес, салоны красоты', lat=55.829593, lng=37.37116
4),
Point(profarea='Транспорт, логистика', lat=53.899117, lng=27.524919),
Point(profarea='Административный персонал', lat=56.830822, lng=60.597501),
Point(profarea='Информационные технологии, интернет, телеком', lat=55.785979, lng=37.66
```

```

0521),
Point(profarea='Искусство, развлечения, масс-медиа', lat=55.785979, lng=37.660521),
Point(profarea='Маркетинг, реклама, PR', lat=55.785979, lng=37.660521),
Point(profarea='Продажи', lat=50.59399, lng=36.600563),
Point(profarea='Административный персонал', lat=61.643647, lng=50.823454),
Point(profarea='Туризм, гостиницы, рестораны', lat=45.074572, lng=41.941191),
Point(profarea='Административный персонал', lat=45.074572, lng=41.941191),
Point(profarea='Спортивные клубы, фитнес, салоны красоты', lat=56.881502, lng=60.5104
1),
Point(profarea='Административный персонал', lat=59.856829603, lng=30.315072455),
Point(profarea='Начало карьеры, студенты', lat=60.0089569106, lng=30.2589382952),
Point(profarea='Туризм, гостиницы, рестораны', lat=60.0089569106, lng=30.2589382952),
Point(profarea='Домашний персонал', lat=55.659231, lng=37.560835),
Point(profarea='Административный персонал', lat=56.331972, lng=36.733432),
Point(profarea='Медицина, фармацевтика', lat=56.331972, lng=36.733432),
Point(profarea='Производство, сельское хозяйство', lat=55.873203, lng=37.122619),
Point(profarea='Транспорт, логистика', lat=55.631773, lng=37.618031),
Point(profarea='Безопасность', lat=55.775388, lng=37.657817),
Point(profarea='Юристы', lat=55.853718, lng=48.568126),
Point(profarea='Медицина, фармацевтика', lat=59.957782, lng=30.342189),
Point(profarea='Административный персонал', lat=51.535389, lng=46.01815),
Point(profarea='Продажи', lat=51.535389, lng=46.01815),
Point(profarea='Спортивные клубы, фитнес, салоны красоты', lat=51.535389, lng=46.0181
5),
Point(profarea='Транспорт, логистика', lat=55.705645, lng=37.689447),
Point(profarea='Рабочий персонал', lat=55.705645, lng=37.689447),
Point(profarea='Транспорт, логистика', lat=55.794067, lng=49.169098),
Point(profarea='Административный персонал', lat=43.673405, lng=40.297525),
Point(profarea='Административный персонал', lat=44.9984885793, lng=39.0794369012),
Point(profarea='Производство, сельское хозяйство', lat=55.726396, lng=37.381801),
Point(profarea='Рабочий персонал', lat=55.726396, lng=37.381801),
Point(profarea='Туризм, гостиницы, рестораны', lat=53.348763, lng=83.777009),
Point(profarea='Начало карьеры, студенты', lat=47.270275, lng=39.72617),
Point(profarea='Продажи', lat=47.270275, lng=39.72617),
Point(profarea='Юристы', lat=55.775545, lng=37.586077),
Point(profarea='Производство, сельское хозяйство', lat=55.853895, lng=37.517734),
Point(profarea='Административный персонал', lat=47.236788, lng=39.614797),
Point(profarea='Банки, инвестиции, лизинг', lat=55.863328, lng=37.540093),
Point(profarea='Бухгалтерия, управленческий учет, финансы предприятия', lat=55.863328,
lng=37.540093),
Point(profarea='Рабочий персонал', lat=55.879101, lng=38.479811),
Point(profarea='Бухгалтерия, управленческий учет, финансы предприятия', lat=55.73441, l
ng=37.59047),
Point(profarea='Строительство, недвижимость', lat=59.910641, lng=30.267404),
Point(profarea='Начало карьеры, студенты', lat=55.719536, lng=37.628155),
Point(profarea='Туризм, гостиницы, рестораны', lat=55.719536, lng=37.628155),
Point(profarea='Маркетинг, реклама, PR', lat=54.953918, lng=82.878056)]

```

```

In [49]: areas = reduce(lambda l, p: l + [Area(p.profarea, [p])] if not any(list(map(partial(Area

```

```

In [50]: geo_profareas_grouped_df = pd.DataFrame(zip(
    map(attrgetter('profarea'), areas),
    map(attrgetter('count'), areas),
    map(attrgetter('center_lat'), areas),
    map(attrgetter('center_lng'), areas),
), columns=['profarea', 'count', 'center_lat', 'center_lng'])

```

```

In [51]: geo_profareas_grouped_df.head(10)

```

Out[51]:

		profarea	count	center_lat	center_lng
0		Административный персонал	229	59.904215	30.321857
1		Домашний персонал	32	59.981839	30.317089
2		Транспорт, логистика	887	55.724103	37.582520
3		Административный персонал	23	48.348426	44.214197
4		Спортивные клубы, фитнес, салоны красоты	337	55.717553	37.631853
5		Продажи	2771	55.773011	37.838572
6		Бухгалтерия, управленческий учет, финансы пред...	25	51.704838	37.703753
7		Строительство, недвижимость	40	52.520757	39.150968
8		Наука, образование	11	55.558580	49.238237
9		Домашний персонал	8	54.450943	49.326572

In [62]:

```
len(areas)
```

Out[62]:

1848

In [52]:

```
px.scatter_geo(
    geo_profareas_grouped_df,
    lat='center_lat',
    lon='center_lng',
    size='count',
    fitbounds='locations',
    color='profarea',
    center={'lat': 53, 'lon': 83},
    size_max=50,
    labels={'count': 'Количество вакансий'},
    title='Количество вакансий и средняя зарплата относительно города'
)
```

## Количество вакансий и средняя зарплата относительно города



### profarea

- Административный персонал
- Домашний персонал
- Транспорт, логистика
- Спортивные клубы, фитнес, салоны красоты
- Продажи
- Бухгалтерия, управленческий учет, финансы предприятия
- Строительство, недвижимость
- Наука, образование
- Информационные технологии, интернет, телеком
- Рабочий персонал
- Медицина, фармацевтика
- Искусство, развлечения, масс-медиа
- Маркетинг, реклама, PR
- Туризм, гостиницы, рестораны
- Управление персоналом, тренинги
- Автомобильный бизнес
- Начало карьеры, студенты

```
In [53]: all_profareas = reduce(set.union, specs_df.specialization_profarea_names, set())
```

```
In [54]: len(all_profareas)
```

```
Out[54]: 28
```

```
In [55]: {profarea: specs_df[specs_df.specialization_profarea_names.map({profarea}.issubset)]['me  
in all_profareas}
```

```
Out[55]: {'Высший менеджмент': 123710.24584717608,
'Закупки': 62442.23834886817,
'Управление персоналом, тренинги': 65502.94584382872,
'Инсталляция и сервис': 61575.194444444445,
'Спортивные клубы, фитнес, салоны красоты': 51376.46395250212,
'Консультирование': 92617.73353751915,
'Безопасность': 53167.152825836216,
'Домашний персонал': 41872.405241935485,
'Начало карьеры, студенты': 43474.821203244705,
'Производство, сельское хозяйство': 64457.76760048721,
'Добыча сырья': 94832.55590062111,
'Бухгалтерия, управленческий учет, финансы предприятия': 48283.09805153991,
'Строительство, недвижимость': 84771.85741049125,
'Автомобильный бизнес': 72177.20721925133,
'Искусство, развлечения, масс-медиа': 56821.15873015873,
'Туризм, гостиницы, рестораны': 47756.217606707316,
'Наука, образование': 45231.578881987574,
'Страхование': 72418.31764705882,
'Рабочий персонал': 64773.36851851852,
'Маркетинг, реклама, PR': 60350.047784967646,
'Продажи': 49176.52389049481,
'Государственная служба, некоммерческие организации': 52380.44400785855,
'Юристы': 55641.35897435898,
'Банки, инвестиции, лизинг': 57242.99230111206,
'Административный персонал': 48667.45219370861,
'Информационные технологии, интернет, телеком': 92008.00538176925,
'Транспорт, логистика': 67427.33646295663,
'Медицина, фармацевтика': 55659.44958753437}
```

```
In [56]: df_profarea = pd.DataFrame({
    profarea: specs_df[
        specs_df.specialization_profarea_names
            .map({profarea}.issubset)
            ][['mean_salary']].agg(['count', 'sum', 'mean', 'max', 'min'])
        for profarea in all_profareas
    }).T
df_profarea = df_profarea.astype(np.int64).reset_index().rename(columns={'index': 'profarea'}
```

```
In [57]: df_profarea.head(10)
```

	profarea	count	sum	mean	max	min
0	Высший менеджмент	602	74473568	123710	1200000	150
1	Закупки	751	46894121	62442	300000	1
2	Управление персоналом, тренинги	794	52009339	65502	525000	10000
3	Инсталляция и сервис	468	28817191	61575	200000	1300
4	Спортивные клубы, фитнес, салоны красоты	1179	60572851	51376	1001000	3000
5	Консультирование	653	60479380	92617	1650000	12500
6	Безопасность	1734	92191843	53167	338696	1500
7	Домашний персонал	496	20768713	41872	250000	1300
8	Начало карьеры, студенты	8876	385882513	43474	550000	30
9	Производство, сельское хозяйство	4105	264599136	64457	650000	1

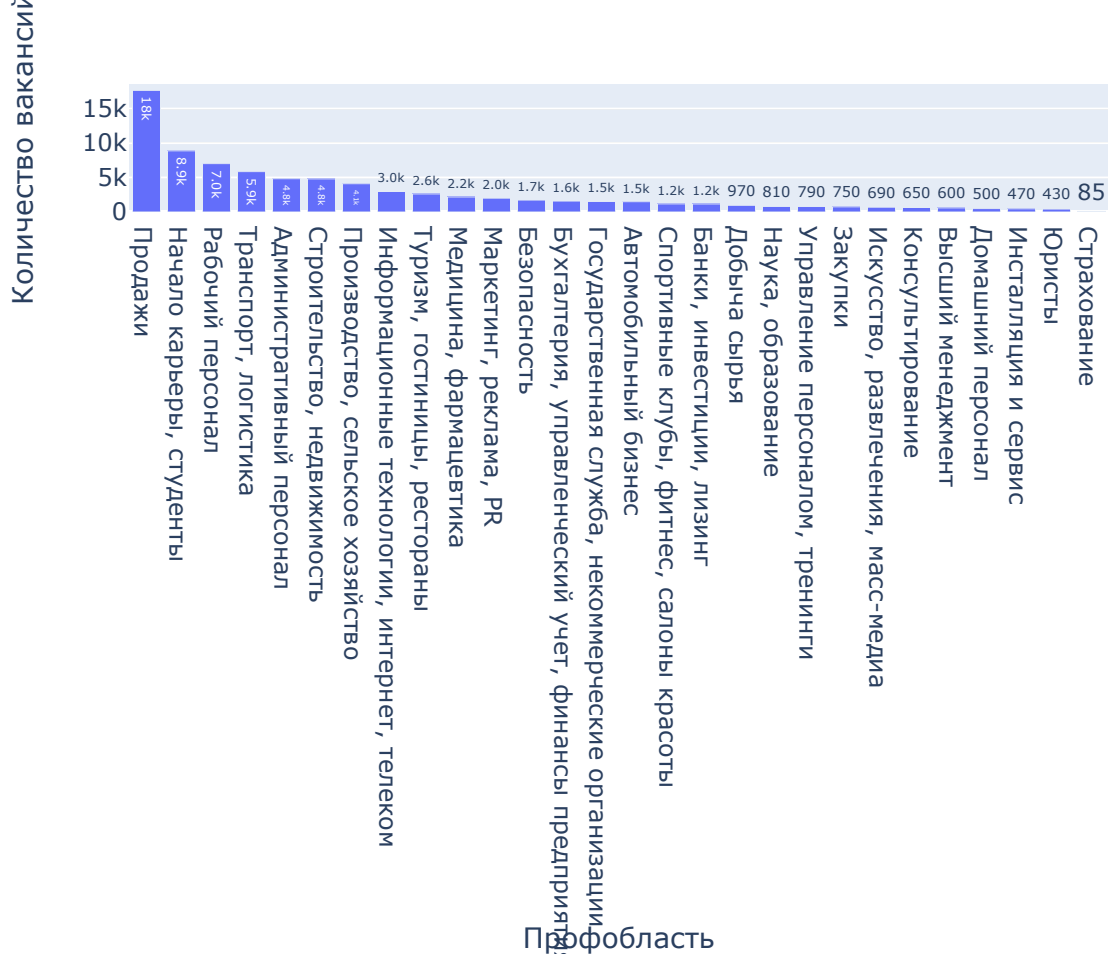
```
In [58]: px.bar(
    df_profarea,
```

```

x='profarea',
y='count',
labels={'profarea': 'Профобласть', 'count': 'Количество вакансий'},
text_auto='.2s',
title='Количество вакансий в каждой области'
).update_xaxes(categoryorder='total descending')

```

Количество вакансий в каждой области



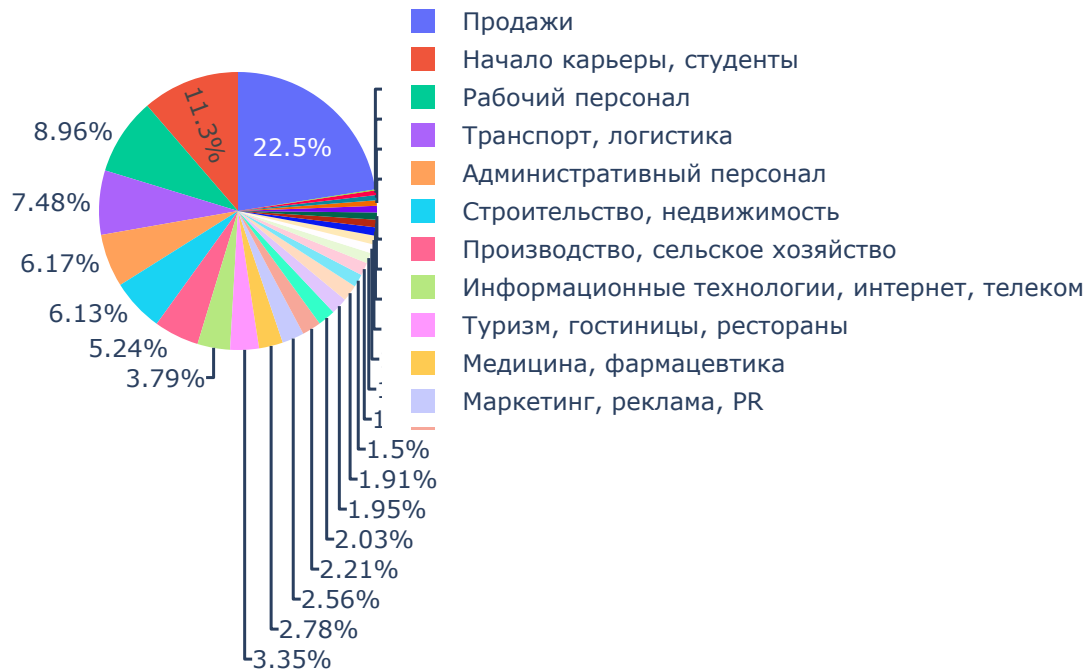
In [59]:

```

px.pie(
    df_profarea,
    names='profarea',
    values='count',
    labels={'index': 'Профобласть', 'count': 'Количество вакансий'},
    title='Доля вакансий для каждой области'
)

```

## Доля вакансий для каждой области

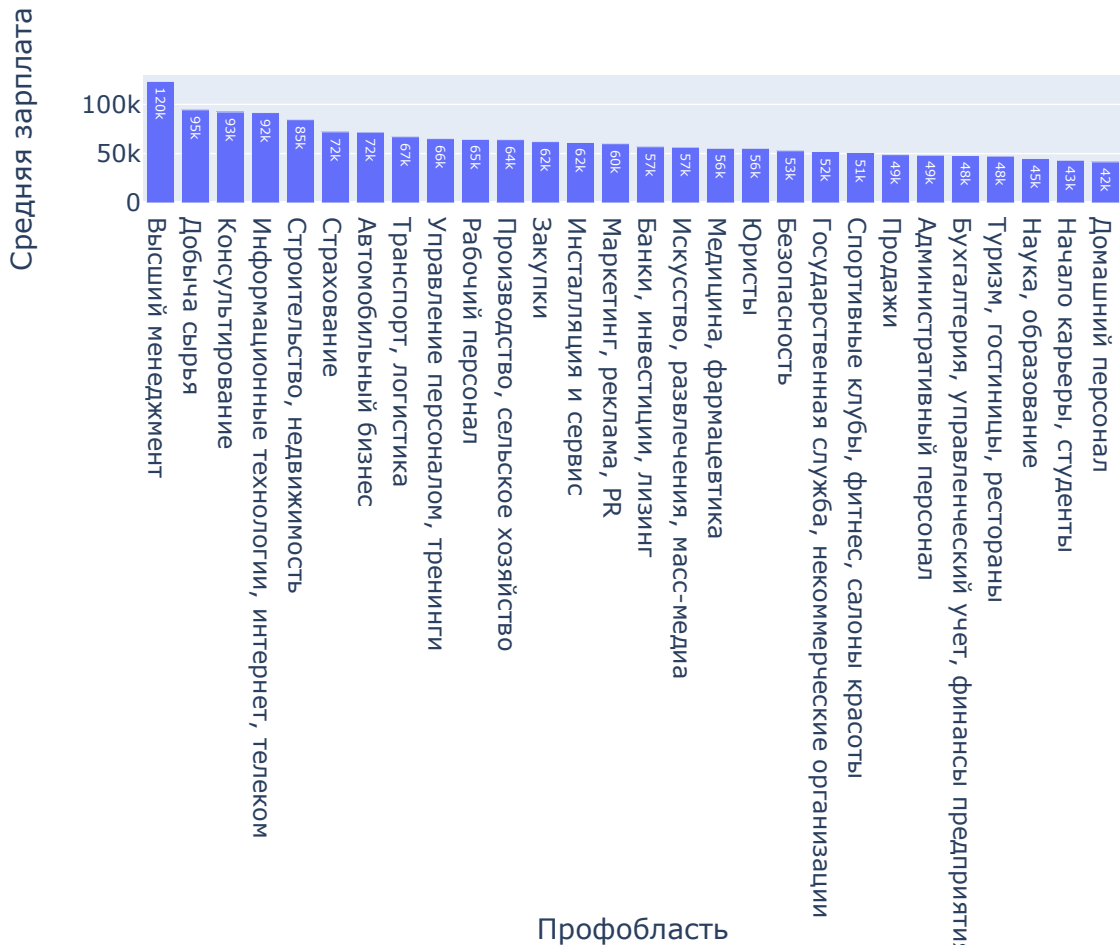


In [60]:

```
px.bar(  
    df_profarea,  
    x='profarea',  
    y='mean',  
    labels={'profarea': 'Профобласть', 'mean': 'Средняя зарплата'},  
    text_auto='.2s',  
    title='Средняя зарплата в каждой области'  
).update_xaxes(categoryorder='total descending')
```



## Средняя зарплата в каждой области



In [61]:

```
px.bar(
    df_profarea,
    x='profarea',
    y='sum',
    labels={'profarea': 'Профобласть', 'sum': 'Сумма всех зарплат'},
    text_auto='.2s',
    title='Сумма зарплат в каждой области'
).update_xaxes(categoryorder='total descending')
```

## Сумма зарплат в каждой области

Сумма всех зарплат

