



```
@app.route('/helfer/stellenangebot', methods=['GET', 'POST']) @login_required def helper_stellenangebot():    form = JobFilterForm()
    categories = Category.query.all()
    form.category.choices = [(0, 'Alle')] + [(c.catId, c.catName) for c in categories]
    jobs = Job.query.filter(Job.statusId ==
1).order_by(Job.date.asc())
    if form.validate_on_submit():
        if form.search.data:
            jobs = jobs.filter(Job.description.ilike(f'%{form.search.data}%'))
        if form.category.data != '0':
            jobs = jobs.filter(Job.catId ==
form.category.data)
        if form.plz.data:
            jobs = jobs.filter(Job.plz == int(form.plz.data))
        if form.min_hours.data:
            jobs = jobs.filter(Job.hours >= form.min_hours.data)
    jobs =
    jobs.paginate(page=request.args.get('page', 1,
type=int), per_page=10)
    return render_template('helfer_stellenangebot.html',
form=form, jobs=jobs).
```

Wenn ich statt Job.query db.session.select(Job)... benutze, bekomme ich den Error: AttributeError: 'ScalarResult' object has no attribute 'paginate'

Der Fehler tritt auf, weil db.session.select(Job) in SQLAlchemy 2.0+ einen Select-Objekt zurückgibt, dessen Ausführung mit db.session.execute() ein ScalarResult erzeugt, und dieses hat keine paginate()-Methode.[\[1\]](#) [\[2\]](#)

Die paginate()-Methode ist eine Erweiterung von Flask-SQLAlchemy und funktioniert primär mit dem legacy Query-Interface (z. B. Job.query) oder in neueren Versionen mit db.paginate(select_statement).^[3] ^[4] ^[5]

Lösung mit modernem Select

Ersetze deinen Code durch diese Variante mit db.paginate() (verfügbar ab Flask-SQLAlchemy 3.0+ für SQLAlchemy 2.0):

```
from sqlalchemy import select # Importiere select

@app.route('/helfer/stellenangebot', methods=['GET', 'POST'])
@login_required
def helfer_stellenangebot():
    form = JobFilterForm()
    categories = Category.query.all()
    form.category.choices = [(0, 'Alle')] + [(c.catId, c.catName) for c in categories]

    # Basis-Select erstellen
    stmt = select(Job).where(Job.statusId == 1).order_by(Job.date.asc())

    if form.validate_on_submit():
        if form.search.data:
            stmt = stmt.where(Job.description.ilike(f'%{form.search.data}%'))
        if form.category.data != '0':
            stmt = stmt.where(Job.catId == form.category.data)
        if form.plz.data:
            stmt = stmt.where(Job.plz == int(form.plz.data))
        if form.min_hours.data:
            stmt = stmt.where(Job.hours >= form.min_hours.data)

    jobs = db.paginate(stmt, page=request.args.get('page', 1, type=int), per_page=10)
    return render_template('helfer_stellenangebot.html', form=form, jobs=jobs)
```

Im Template iterierst du über jobs.items für die Job-Objekte.^[4] ^[5]

Alternative: Manuelle Pagination

Falls db.paginate() nicht verfügbar ist, implementiere sie manuell mit limit() und offset():

```
# Nach den Filtern:
page = request.args.get('page', 1, type=int)
per_page = 10
offset = (page - 1) * per_page
total = db.session.scalar(select(func.count()).select_from(stmt.subquery()))

jobs_query = db.session.execute(stmt.limit(per_page).offset(offset)).scalars().all()
```

Berechne total_pages = (total + per_page - 1) // per_page und passe das Template an.^[6]

1. <https://stackoverflow.com/questions/18409645/sqlalchemy-does-not-work-with-pagination>
2. <https://flask-sqlalchemy.readthedocs.io/en/stable/queries/>
3. https://koor.fr/Python/API/web/flask_sqlalchemy/SQLAlchemy/paginate.wp
4. <https://flask-sqlalchemy.readthedocs.io/en/stable/pagination/>
5. <https://github.com/pallets-eco/flask-sqlalchemy/blob/main/docs/pagination.rst>
6. <https://iifx.dev/en/articles/62450266>
7. <https://flask-sqlalchemy.readthedocs.io/en/stable/api/>
8. <https://uriyyo-fastapi-pagination.netlify.app/integrations/sqlalchemy/paginate/>
9. <https://stackoverflow.com/questions/9916094/sqlalchemy-and-going-through-a-large-result-set>
10. https://www.reddit.com/r/flask/comments/ny1lal/sqlalchemy_and_pagination/
11. https://flask-sqlalchemy.readthedocs.io/en/stable/legacy_query/
12. https://www.reddit.com/r/flask/comments/3nsfr3/afflasksqlalchemy_pagination/
13. <https://gist.github.com/jas-haria/a993d4ef213b3c0dd1500f86d31ad749>
14. <https://stackoverflow.com/questions/43103585/python-flask-sqlalchemy-pagination>
15. <https://www.digitalocean.com/community/tutorials/how-to-query-tables-and-paginate-data-in-flask-sqlalchemy>
16. <https://stackoverflow.com/questions/64835274/can-flask-sqlalchemy-s-paginate-class-be-used-with-a-python-list>
17. <https://github.com/uriyyo/fastapi-pagination/issues/191>
18. <https://www.youtube.com/watch?v=hkL9pgCJPNk>
19. <https://stackoverflow.com/questions/72790215/sqlalchemy-2-x-with-specific-columns-makes-scalars-return-non-orm-objects>
20. <https://www.youtube.com/watch?v=U18hO1ngZEQ>
21. <https://www.digitalocean.com/community/tutorials/how-to-perform-flask-sqlalchemy-migrations-using-flask-migrate>
22. <https://www.youtube.com/watch?v=WND3gNDtMWQ>
23. <https://flask-sqlalchemy.readthedocs.io/en/stable/changes/>
24. <https://www.youtube.com/watch?v=uNmWxvvyBGU>
25. https://www.reddit.com/r/flask/comments/lpcdbk/how_do_i_paginate_a_mysql_query/
26. <https://github.com/pallets-eco/flask-sqlalchemy/issues/1168>
27. <https://github.com/sqlalchemy/sqlalchemy/issues/6333>
28. <https://stackoverflow.com/questions/15727155/how-to-paginate-in-flask-sqlalchemy-for-db-session-jointed-queries>