# steps for dumping VACUUM sqlite3

after vacuuming, database integrity is compromised. so we need to use .recover to fix the dump before dumping in json for migration.

## 1. check for issues with database using `pragma integrity_check`

```
~/evco_os$ sqlite3 db.sqlite3 "pragma integrity_check"
```

example problematic output:

```
~/evco_os$ sqlite3 db.sqlite3 "pragma integrity_check"

*** in database main ***
On tree page 161588 cell 0: invalid page number 161867
On tree page 161588 cell 422: invalid page number 161866
On tree page 161588 cell 421: invalid page number 161864
On tree page 161588 cell 420: invalid page number 161862
On tree page 161588 cell 419: invalid page number 161861
On tree page 161588 cell 418: invalid page number 161859
On tree page 161588 cell 417: invalid page number 161858
On tree page 161588 cell 416: invalid page number 161856
On tree page 161588 cell 415: invalid page number 161855
On tree page 161588 cell 414: invalid page number 161854
On tree page 161588 cell 413: invalid page number 161852
On tree page 161588 cell 412: invalid page number 161851
On tree page 160588 cell 0: invalid page number 161865
On tree page 160588 cell 417: invalid page number 161860
On tree page 160588 cell 416: invalid page number 161857
On tree page 160588 cell 415: invalid page number 161853
Error: database disk image is malformed
```

example ok output:

```
~/evco_os$ sqlite3 db.sqlite3 "pragma integrity_check"

ok
```

## 2. create new db via `.dump/.recover`

if you have problems, try:

```
~/evco_os$ sqlite3 db.sqlite3 ".recover" | sqlite3 new_db.sqlite3
```

else:

```
~/evco_os$ sqlite3 db.sqlite3 ".dump" | sqlite3 new_db.sqlite3
```

## 3. migrate your database

shell:

```
~/evco_os$ python manage.py makemigrations
~/evco_os$ python manage.py migrate
```

## 4. dump out

with StatusLog

```
python manage.py dumpdata --natural-foreign --natural-primary -e contenttypes -e auth.Permission --indent 4 > prod_db_dump.json
```

without StatusLog (if logs causing error)

```
python manage.py dumpdata --natural-foreign --natural-primary -e contenttypes -e auth.Permission --indent 4 > prod_db_dump.json
```

# references

1. dumping and recovering sqlite
   - How to recover a corrupt SQLite3 database
   - .dump creating empty db file
2. corrupted db
   - malformed db
3. safe dumpdata in django
   - Dump Your Django Database and Load It into a New Project