

Five Stars and What They Hide

A Relational View of Food Hygiene Ratings Across Seven Authorities

Declaration of code and sources

The database design, loader, SQL queries, E/R diagrams, Express routes, EJS views and CSS were written for this project. Files containing student-written logic are labelled with an original work comment. Where framework patterns were adapted from documentation, the relevant file names the source, for example, the Express error-handling guide and the mysql2 pool documentation.

Contents

- 1. Dataset choice and critique
- 2. E/R and relational modelling
- 3. MySQL implementation and query results
- 4. Node web application
- 5. Referencing, originality and extra credit
- References
- Appendix A. Create commands
- Appendix B. SQL commands for the research questions
- Appendix C. Application screenshots

Summary

This project transforms the Food Standards Agency (FSA) Food Hygiene Rating Scheme (FHRS) open data into a relational MySQL database and a Node web application. The dataset contains 21,209 establishments from seven local authorities across England, Wales and Scotland. Six authorities use the numeric Food Hygiene Rating Scheme, while Shetland Islands uses the pass-based Food Hygiene Information Scheme.

The design separates authorities, regions, business types, rating schemes and rating values into related tables. Inspection stores the current rating snapshot and its available sub-scores. The structure supports authority summaries, postcode searches, business-type comparisons, rating distributions and establishment details. The decomposition removes the main repeated and transitive dependencies in a flat source file. Six relations satisfy Boyce-Codd normal form. Establishment retains one documented dependency involving scheme_id, with a cleaner alternative described in the normalisation analysis. Each research question requires joins, filtering, grouping or aggregation across related tables.

1. Dataset choice and critique

1.1 Dataset and interest

I selected the Food Standards Agency Food Hygiene Rating Scheme open data. Local authority officers inspect food businesses and publish the current rating for each establishment. England and Wales use the FHRS scale from 0 to 5, where 5 means very good. Scotland uses FHIS, a pass and improvement scheme. The project uses six FHRS authorities and Shetland Islands as an FHIS authority, allowing the model to accommodate two published rating schemes within one structure (Food Standards Agency, 2026a).

The topic is interesting because it links public health, local government and everyday consumer choice. The data had enough structure to be worth modelling: an establishment belongs to an authority, has a business type, is governed by a scheme, receives a rating value, and has an inspection snapshot with sub-scores where the FHRS scheme provides them.

1.2 Sources and scope

Source	Use in project	Licence	
Food Standards Agency FHRS open data	Primary source for establishments, authority details, ratings, rating dates, addresses, business types, sub scores and geocodes.	Open Government Licence v3.0	
FSA per-authority XML files	Seven XML files loaded for Birmingham, Cardiff, Shetland Islands, Bath and North East Somerset, York, Brighton and Hove, and Rutland.	Open Government Licence v3.0	
ONS Open Geography Portal	Region mapping used to connect each local authority to a region or country label for regional grouping.	Open Government Licence v3.0	
Authority	Region	Scheme	Establishments
Birmingham	West Midlands	FHRS	10,037
Cardiff	Wales	FHRS	3,435
Brighton and Hove	South East	FHRS	3,011
York	Yorkshire and The Humber	FHRS	2,149
Bath and North East Somerset	South West	FHRS	1,841
Rutland	East Midlands	FHRS	458
Shetland Islands	Scotland	FHIS	278

The FSA files provide the primary data, while the ONS source supplies the regional mapping used for grouping authorities (Food Standards Agency, 2026a; Office for National Statistics, 2026).

1.3 Dataset critique

Criterion	Assessment
Quality	The data is authoritative because the inspections are carried out by local authorities and published through the FSA. The main quality issue is missing optional fields, especially postcodes, geocodes and sub scores for FHIS records. These are stored as NULL, preserving the difference between an absent value and a score of zero.
Level of detail	The data is at establishment level. It includes address fields, business type, authority, rating value, rating date, a pending flag, geocode, and FHRS sub-scores for hygiene, structure and confidence in management. This level supports aggregation and record-level lookup.
Documentation	The FSA portal documents the API and per-authority file structure. Rating meanings are published with the scheme. The region lookup is separately documented by the ONS.
Interrelation	The source contains repeated authority and business type values that map naturally into related tables. The model also links to an ONS region lookup. Geocodes could support later mapping, outside the scope of this report.
Use	The data can support a public-facing search tool, authority comparison, council reporting and consumer awareness. The questions chosen here are descriptive and auditable.
Discoverability	The FSA publishes the data on a public open data page with per-authority XML files and an API. This makes retrieval straightforward and repeatable.

The sources are released under the Open Government Licence v3.0, which permits copying, adaptation and reuse with attribution. The licence provides no warranty for the accuracy or completeness of the information. Missing values and limitations must therefore be reported openly. The report, loader, README and web application identify the FSA and ONS sources (The National Archives, n.d.).

1.4 Research questions

No.	Question	Reason a database fits
1	Which authorities have the highest and lowest share of establishments rated 5, and does the pattern hold within a business type?	Requires joins between authority, establishment, inspection, rating value and business type, then grouped aggregation.
2	For a chosen postcode area, which establishments hold a rating below 3, and what are their hygiene, structural and confidence sub scores?	Requires filtering, a postcode prefix search, record retrieval and joined inspection data.
3	How do mean hygiene, structural and confidence scores compare across business types, given that a lower score is better?	Requires grouped averages over inspection and business type records.
4	Within a selected authority, what share of establishments is awaiting inspection or has a new rating pending?	Requires status values and flags across rating and inspection tables.
5	What is the most common business type in each authority, and among Restaurant, Cafe and Canteen premises, where are ratings weakest?	Requires grouped counts to identify the most common type in each authority, followed by joins and grouped averages to compare one fixed business type across authorities.
6	Comparing two authorities side by side, how do their rating distributions differ across the 0 to 5 scale?	Requires grouping by rating value for two selected authorities.

These questions are descriptive database questions. A single spreadsheet would make repeated sorting and copying likely, because authority, scheme, rating value and business type data are repeated across rows. A statistical or machine learning model would add unnecessary complexity because the goal is retrieval, comparison and aggregation, not prediction.

2. E/R and relational modelling

2.1 Conceptual E/R model

The conceptual model uses Chen notation: rectangles for entities, ellipses for attributes, diamonds for relationships, underlined primary keys, a dashed ellipse for the derived overall_score attribute and double lines for total participation. These conventions follow the notation used in the module. The seven entities are Region, LocalAuthority, BusinessType, Scheme, RatingValue, Establishment and Inspection.

Establishment is central because it holds the premises' details. Inspection is separate because the rating date, sub-scores, rating value and pending flag describe an assessment snapshot. RatingValue is separate from Scheme so FHRS numeric ratings and FHS pass or improvement values can coexist in one model. The dashed overall_score attribute represents a project-specific sum of the three available sub-scores. It is calculated when required and is not stored. This measure is not the published FSA hygiene rating.

FHRS conceptual E/R model, Chen notation

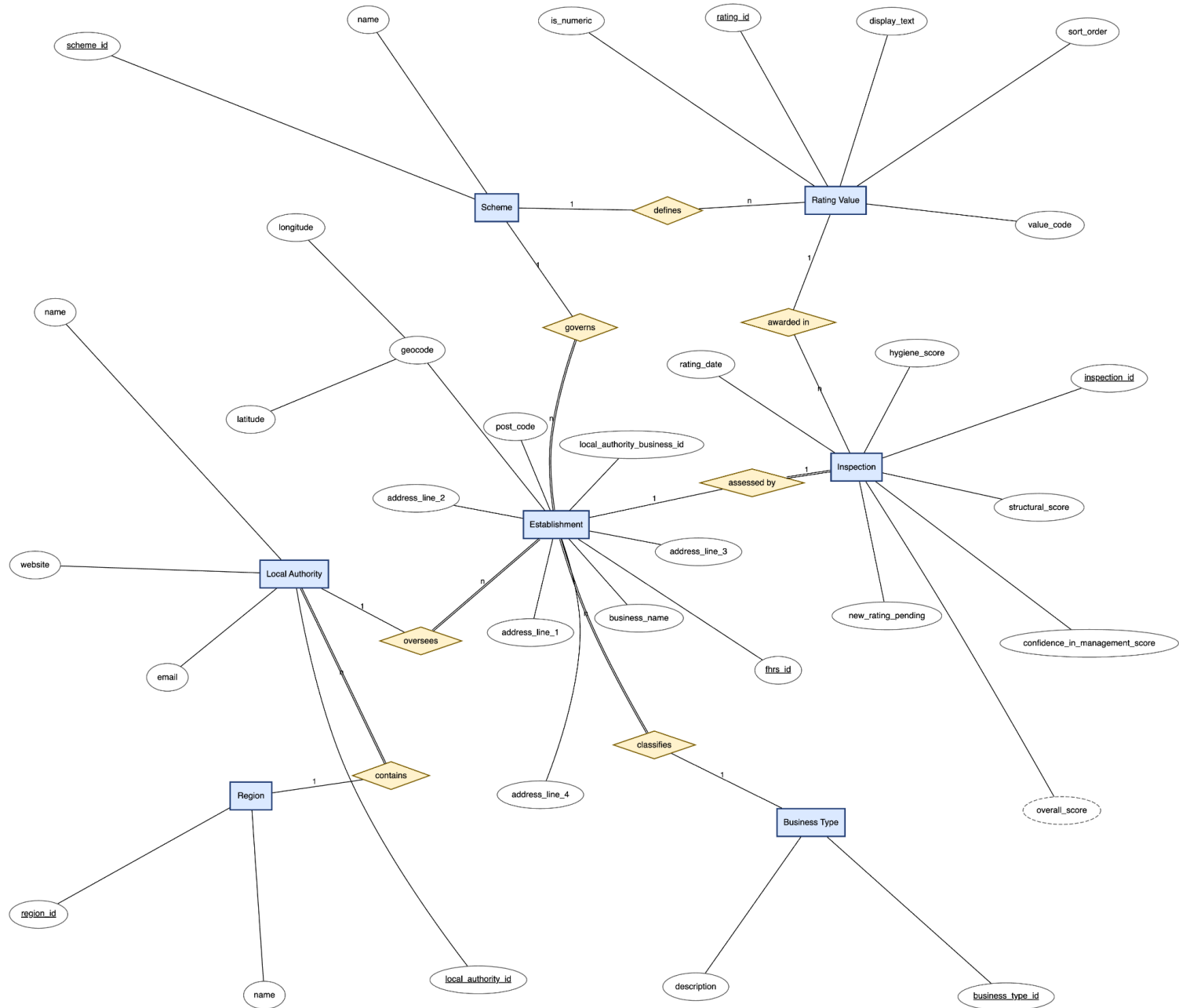


Figure 1. Conceptual E/R model in Chen notation

FHRS relational mapping

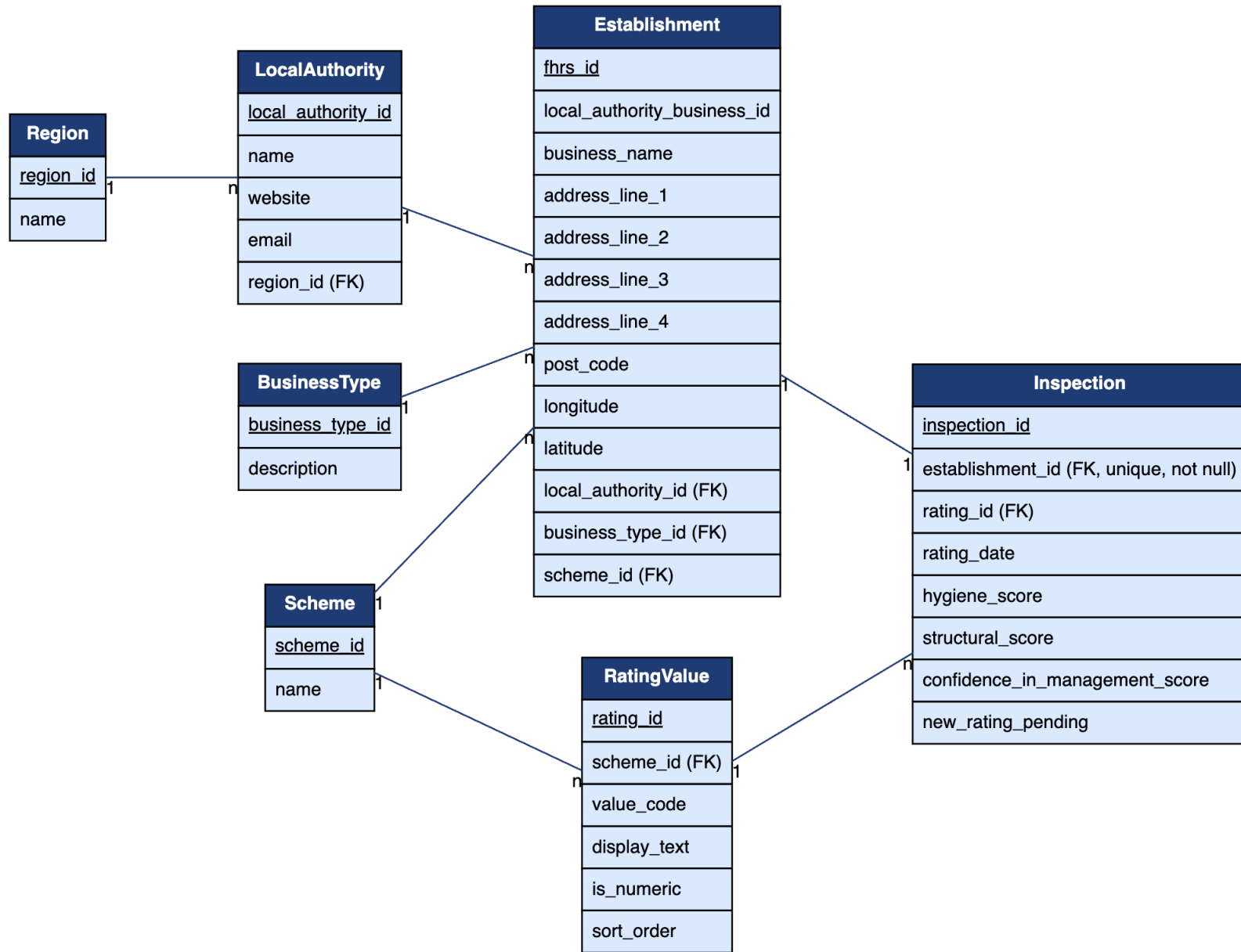


Figure 2. Relational mapping with primary keys, foreign keys and cardinalities

2.2 Cardinality and mapping decisions

Relationship	Cardinality	Relational implementation
Region contains LocalAuthority	One-to-many	local_authority.region_id
LocalAuthority oversees Establishment	One-to-many	establishment.local_authority_id
BusinessType classifies Establishment	One-to-many	establishment.business_type_id
Scheme governs Establishment	One-to-many	establishment.scheme_id
Scheme defines RatingValue	One-to-many	rating_value.scheme_id
Establishment is assessed by Inspection	One-to-one	inspection.establishment_id is unique and not null
RatingValue is awarded in Inspection	One-to-many	inspection.rating_id

The relational mapping places each foreign key on the many side. The composite geocode attribute is flattened into longitude and latitude, which gives atomic columns. The derived overall score is calculated from the three sub-scores and is not stored, reducing the risk of inconsistent derived data.

The source snapshot provides one current inspection record for each loaded establishment. The unique and not null foreign key on inspection.establishment_id ensures that each inspection refers to one establishment and that an establishment can have no more than one stored inspection. The constraint does not require every establishment to have an inspection row. A validation query confirms that the loaded instance contains one inspection for every establishment. A historical extension could remove the unique constraint and store several dated inspections for each establishment.

2.3 Tables and normalisation

Table	Main fields	Purpose
region	region_id, name	Stores each region label once.
scheme	scheme_id, name	Separates FHRS and FHIS.
business_type	business_type_id, description	Removes repeated business type descriptions.
local_authority	local_authority_id, name, website, email, region_id	Stores authority details once and links them to a region.
rating_value	rating_id, scheme_id, value_code, display_text, is_numeric, sort_order	Stores possible ratings per scheme.
establishment	fhirs_id, authority business id, name, address, postcode, geocode, foreign keys	Stores premises details.
inspection	inspection_id, establishment_id, rating_id, rating_date, sub scores, pending flag	Stores the current rating snapshot.

The schema satisfies first normal form because every column stores one atomic value and there are no repeating groups. The conceptual geocode is divided into longitude and latitude. The four address lines are fixed source fields. Second normal form is satisfied because each non-key attribute depends on the whole of every candidate key. The composite alternate key (scheme_id, value_code) in RatingValue determines the complete row, with no partial dependency.

The principal functional dependencies are:

```
region_id -> region.name
scheme_id -> scheme.name
business_type_id -> business_type.description
local_authority_id -> local_authority.name, website, email, region_id
rating_id -> rating_value.attributes
(scheme_id, value_code) -> rating_value.attributes
fhirs_id -> establishment.attributes
inspection_id -> inspection.attributes
establishment_id -> inspection.attributes
```

Region and Scheme each have a unique name, so the name is an alternate candidate key. RatingValue has the candidate keys rating_id and (scheme_id, value_code). Inspection has the candidate keys inspection_id and establishment_id because establishment_id is unique.

Separating authority details and business-type descriptions removes transitive dependencies that would occur in a flat establishment table. One example is fhirs_id -> local_authority_id -> authority name. Region, Scheme, BusinessType, LocalAuthority, RatingValue and Inspection satisfy Boyce-Codd normal form because every determinant in those relations is a candidate key.

Establishment contains one documented exception. Every authority in the loaded data operates one scheme, so the instance satisfies `local_authority_id -> scheme_id`. Both columns are non-key attributes of Establishment. This creates the transitive dependency `fhrs_id -> local_authority_id -> scheme_id`, which leaves Establishment below strict third normal form on `scheme_id`.

A cleaner design would place `scheme_id` in LocalAuthority and obtain the scheme through `local_authority_id`. This removes repeated scheme values and prevents an establishment from carrying a scheme that conflicts with its authority.

The submitted design retains the column because changing it would affect the schema, loader, diagrams and application queries. The loader assigns one scheme consistently to each authority inside a controlled transaction, keeping the current instance coherent. The dependency remains a structural consistency risk and is recorded as a limitation.

No relation contains independent multivalued facts that require a separate fourth-normal-form decomposition. A schema-wide fourth-normal-form claim is not made because Establishment does not satisfy Boyce-Codd normal form on the retained `scheme_id` attribute.

3. MySQL implementation and query results

3.1 Schema implementation

The schema is implemented in MySQL using InnoDB and utf8mb4. Source identifiers are used as primary keys where reliable values exist, such as FHRSID for establishment and LocalAuthorityCode for local_authority. The rating_value table uses a generated rating_id and a unique constraint on scheme_id with value_code, because the same rating text pattern can occur across schemes.

Implementation feature	How it is used
Primary keys	Every table has a primary key. Lookup tables use source keys where available.
Foreign keys	Every relationship in the relational model is enforced with a foreign key.
Indexes	Foreign keys are indexed. post_code has a separate index for prefix searches such as BN1.
CHECK constraints	Sub scores must be 0 to 30 when present. Longitude and latitude must fall in valid ranges. The pending flag must be 0 or 1. Rating dates must be at least 2000-01-01 when present.
One-to-one inspection	inspection establishment_id is unique and not null, so each inspection belongs to exactly one establishment and each establishment can have at most one stored current inspection.
Least privilege	fhrs_app has SELECT only for the web application. fhrs_build has the write privileges needed by the loader.

The full CREATE commands are included in Appendix A. Passwords are placeholders in create_user.sql and should be changed locally. The submitted repository does not include the real .env file.

3.2 Loading and cleaning the data

The loader downloads the seven per-authority XML files, caches them locally, parses them with fast-xml-parser and loads them through mysql2 (fast-xml-parser, n.d.; mysql2, n.d.). Each authority is loaded inside one transaction, so a failed file leaves no partial authority data. INSERT ... ON DUPLICATE KEY UPDATE makes the loading operation idempotent. Re-running the loader updates existing records without creating duplicates. This was checked by running the loader twice and comparing the table counts.

Source issue	Loading decision
Blank text	Stored as NULL.
Postcodes	Upper case with single spaces.
NewRatingPending	Mapped from True or False text to 1 or 0.
RatingDate	Nil or empty values stored as NULL. Valid source dates kept as YYYY-MM-DD.
Sub scores	Missing values stored as NULL, preserving zero as a real score.

Geocode	Missing longitude or latitude stored as NULL.
FHIS rating values	Spaced text such as Improvement Required is seeded exactly so the lookup resolves correctly.
Table	Rows after load
region	7
scheme	2
business_type	14
local_authority	7
rating_value	15
establishment	21,209
inspection	21,209

3.3 Reflection on the database fit

The decomposition works well for the dataset. Authority details, region names, business-type descriptions and rating labels are stored once, improving consistency and making the research queries easier to express. The same structure accommodates FHRS and FHIS ratings, supporting the decision to model Scheme and RatingValue separately. A validation query confirms the loaded one-to-one condition: a left join from Establishment to Inspection returns zero establishments without an inspection row.

The source contains only the current inspection snapshot. The database cannot examine rating changes over time. A historical design would permit several Inspection rows for each Establishment and use the inspection date to distinguish them. FHIS records have no numeric ratings or component sub-scores, so score-based questions apply only to FHRS records. Postcode remains an establishment attribute because it does not reliably determine a local authority. Wales and Scotland are stored in the single analytical Region relation to support consistent grouping in this assessment.

Missing values are preserved as NULL. Approximately 2,400 premises have no postcode and approximately 3,400 have no geocode in the loaded snapshot. Retaining these absences avoids fabricated precision and keeps a missing value distinct from a recorded value of zero.

3.4 Query results

Question	Result
1	Rutland has the highest share rated 5 at 84.6 per cent. Birmingham has the lowest at 65.9 per cent. Within Restaurant, Cafe and Canteen, Brighton and Hove leads at 80.4 per cent, and Birmingham remains lowest at 62.2 per cent.
2	The BN1 postcode area returns 18 premises rated below 3. The sub scores explain the low ratings, including cases with hygiene scores as high as 20 on a scale where a lower score is better.
3	Takeaway or sandwich shop has the highest mean hygiene, structural and confidence scores, making it the weakest group on average. Schools and farmers have the lowest mean scores.
4	For Birmingham, 1,620 establishments are awaiting inspection, and 124 have a new rating pending. The combined share is 17.2 per cent.
5	Restaurant, Cafe and Canteen is the most common type in five authorities. Other catering premises is most common in Rutland, while Retailers, other is most common in Shetland Islands. The second query compares Restaurant, Cafe and Canteen because it is the most common type overall. Birmingham has the weakest mean numeric rating for this type at 4.19, while Brighton and Hove has the strongest at 4.71.
6	Birmingham and Cardiff are both concentrated at rating 5. Cardiff has a larger rated-5 share, at 69.8 per cent compared with Birmingham at 65.9 per cent. Birmingham has larger proportions at ratings 0, 1 and 2. Both authorities have 8.5 per cent at rating 3. Birmingham has more establishments, which explains its larger raw counts.

The SQL commands answering the research questions are included in Appendix B. They use joins and aggregation across at least three tables, matching the database design and the motivation set out in Stage 1.

4. Node web application

4.1 Architecture

The web application uses Node, Express, EJS and the mysql2 promise API (Express, n.d.; EJS, n.d.; mysql2, n.d.). It is divided into a data-access layer, route handlers and server-rendered views. One shared connection pool is created when the application starts and reused for all database operations. The application connects through `fhrrs_app`, which has SELECT privileges only.

Area	Role
<code>src/db.js</code>	Creates one mysql2 promise pool using environment variables.
<code>src/queries.js</code>	Holds parameterised query functions for authorities, search, details, summaries, distributions and comparisons.
<code>src/app.js</code>	Defines routes, validates IDs, renders pages, returns JSON and handles errors centrally.
<code>views</code>	Contains EJS templates with semantic HTML, form labels and accessible tables.
<code>public/style.css</code>	Styles layout, tables, filters, navigation and the server-rendered bar chart.

User input is validated before it reaches the database. Identifiers and page numbers are converted to positive integers. Minimum and maximum ratings are checked against the range 0 to 5, allowing a valid value of 0. The postcode-area input is restricted to letters, digits and spaces, preventing SQL LIKE wildcard characters from entering the filter. Every value is passed to a parameterised prepared statement in `queries.js`.

Search results are paginated at 25 records per page. The query requests one additional record so the application can determine whether another page exists without assuming that a full page guarantees further results.

The online lab environment serves applications behind a configurable path prefix. A small `urlFor` helper adds `BASE_PATH` to internal links, form actions and the stylesheet. An empty value is used when the application runs locally.

4.2 Pages and relation to the questions

Route	Purpose	Research link
<code>/</code>	Home page with data description, licence attribution and research questions.	Frames all questions.
<code>/establishments</code>	Searches by authority, business type, postcode area, minimum rating and maximum rating. Results are paginated.	Supports questions 1 and 2. A postcode area with a maximum rating of 2 lists premises rated below 3.
<code>/establishments/:id</code>	Full establishment details with sub scores, pending flag, overall score and geocode.	Supports question 2 at record level.
<code>/authorities/:id</code>	Authority summary with counts, score averages and rating distribution chart.	Supports questions 1, 3 and 4.
<code>/compare?a=<id>&b=<id></code>	Two authorities compared across the numeric scale.	Supports question 6.
<code>/api/authorities/:id/summary</code>	Read-only JSON summary and rating distribution.	Shows the same data can be reused by another client.

4.3 Accessibility, validation and error handling

The EJS templates use semantic landmarks, ordered headings, a skip link, labelled form controls, table captions and scoped table headers. The rating chart is accompanied by a table containing the same values, so the information does not depend on the visual bars. Invalid identifiers return a clear 400 response, and missing records return 404. Unhandled faults pass to a central error handler. The server logs the technical error while the browser receives a generic 500 message, preventing database details from appearing on the page. Rating distributions and comparison tables list every numeric rating from 0 to 5, including categories with a count of zero. Missing scores, dates, postcodes and geocodes are displayed as Not provided, keeping absent data distinct from a genuine value of zero.

4.4 Application screenshots

The principal application screens are reproduced in Appendix C. They demonstrate the dataset description, search filters, authority summaries, FHIS support, establishment details, authority comparison and the read-only JSON endpoint.

5. Referencing, originality and extra credit

5.1 Referencing and code labelling

The report cites the FSA data source, the ONS mapping, the Open Government Licence and the main framework documentation. Source attribution also appears in the loader, schema comments, README and home page. Student-written files carry original work comments. Adapted patterns identify their source, including the Express error-handling guide, mysql2 pool documentation and EJS documentation (Express, n.d.; mysql2, n.d.; EJS, n.d.).

5.2 Extra features

Feature	Value added
Transactions in the loader	Each authority file loads as one unit of work.
Least privilege users	The app can only SELECT. The build user has write privileges for loading.
Indexing and EXPLAIN work	Foreign key indexes support joins, and the postcode index changes a postcode prefix query from a full scan to a range scan.
Data cleaning and alignment	The loader standardises postcodes, booleans, dates and missing values.
Linked second dataset	The ONS region mapping allows regional grouping.
One model, two schemes	FHRS and FHIS records load through the same schema.
Accessible web output	The chart has a table alternative, and the pages use accessible structure.
Read-only JSON endpoint	The application can serve the authority summary in machine-readable form.

References

dotenv (n.d.) *dotenv package documentation*. Available at: <https://www.npmjs.com/package/dotenv> (Accessed: 9 July 2026).

EJS (n.d.) *Embedded JavaScript templating documentation*. Available at: <https://ejs.co/> (Accessed: 9 July 2026).

Express (n.d.) *Error handling*. Available at: <https://expressjs.com/en/guide/error-handling.html> (Accessed: 9 July 2026).

fast-xml-parser (n.d.) *fast-xml-parser package documentation*. Available at: <https://www.npmjs.com/package/fast-xml-parser> (Accessed: 9 July 2026).

Food Standards Agency (2026a) *Food Hygiene Rating Scheme open data*. Available at: <https://ratings.food.gov.uk/open-data> (Accessed: 9 July 2026).

Food Standards Agency (2026b) *FHRS per-authority XML files*. Data snapshot downloaded 29 June 2026. Available at:
<https://ratings.food.gov.uk/OpenDataFiles/FHRS402en-GB.xml>;
<https://ratings.food.gov.uk/OpenDataFiles/FHRS556en-GB.xml>;
<https://ratings.food.gov.uk/OpenDataFiles/FHRS789en-GB.xml>;
<https://ratings.food.gov.uk/OpenDataFiles/FHRS857en-GB.xml>;
<https://ratings.food.gov.uk/OpenDataFiles/FHRS868en-GB.xml>;
<https://ratings.food.gov.uk/OpenDataFiles/FHRS875en-GB.xml>;
<https://ratings.food.gov.uk/OpenDataFiles/FHRS879en-GB.xml> (Accessed: 29 June 2026).

mysql2 (n.d.) *Documentation for the Node MySQL client and promise API*. Available at:
<https://sidorares.github.io/node-mysql2/docs> (Accessed: 9 July 2026).

Office for National Statistics (2026) *Open Geography Portal*. Available at: <https://geoportal.statistics.gov.uk/> (Accessed: 9 July 2026).

The National Archives (n.d.) *Open Government Licence for public sector information, version 3.0*. Available at:
<https://www.nationalarchives.gov.uk/doc/open-government-licence/version/3/> (Accessed: 9 July 2026).

Appendix A. Create commands

These commands reproduce the database structure and database users. Passwords shown here are placeholders from create_user.sql.

```
DROP DATABASE IF EXISTS fhrrs;
CREATE DATABASE fhrrs CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
USE fhrrs;
```

```
CREATE TABLE region (
  region_id  SMALLINT UNSIGNED NOT NULL,
  name       VARCHAR(100) NOT NULL,
  PRIMARY KEY (region_id),
  UNIQUE KEY uq_region_name (name)
) ENGINE=InnoDB;
```

```
CREATE TABLE scheme (
  scheme_id  TINYINT UNSIGNED NOT NULL,
  name       VARCHAR(20) NOT NULL,
  PRIMARY KEY (scheme_id),
  UNIQUE KEY uq_scheme_name (name)
) ENGINE=InnoDB;
```

```
CREATE TABLE business_type (
  business_type_id  INT UNSIGNED NOT NULL,
  description        VARCHAR(100) NOT NULL,
  PRIMARY KEY (business_type_id)
) ENGINE=InnoDB;
```

```
CREATE TABLE local_authority (
  local_authority_id  SMALLINT UNSIGNED NOT NULL,
  name                VARCHAR(100) NOT NULL,
  website             VARCHAR(255) NULL,
  email               VARCHAR(255) NULL,
  region_id           SMALLINT UNSIGNED NOT NULL,
  PRIMARY KEY (local_authority_id),
  KEY idx_local_authority_region (region_id),
  CONSTRAINT fk_local_authority_region
    FOREIGN KEY (region_id) REFERENCES region (region_id)
    ON DELETE SET NULL ON UPDATE CASCADE
) ENGINE=InnoDB;
```

```
CREATE TABLE rating_value (
  rating_id      SMALLINT UNSIGNED NOT NULL AUTO_INCREMENT,
  scheme_id      TINYINT UNSIGNED NOT NULL,
  value_code     VARCHAR(30) NOT NULL,
  display_text   VARCHAR(80) NOT NULL,
  is_numeric     TINYINT(1) NOT NULL,
  sort_order     TINYINT NOT NULL,
  PRIMARY KEY (rating_id),
  UNIQUE KEY uq_rating_scheme_code (scheme_id, value_code),
  CONSTRAINT fk_rating_scheme
    FOREIGN KEY (scheme_id) REFERENCES scheme (scheme_id)
    ON DELETE RESTRICT ON UPDATE CASCADE,
  CONSTRAINT chk_rating_is_numeric CHECK (is_numeric IN (0, 1))
) ENGINE=InnoDB;
```

```
CREATE TABLE establishment (
  fhrrs_id          INT UNSIGNED NOT NULL,
  local_authority_business_id  VARCHAR(100) NULL,
  business_name     VARCHAR(255) NOT NULL,
  address_line_1    VARCHAR(200) NULL,
  address_line_2    VARCHAR(200) NULL,
  address_line_3    VARCHAR(200) NULL,
```

```

address_line_4          VARCHAR(200) NULL,
post_code               VARCHAR(12) NULL,
longitude               DECIMAL(10,7) NULL,
latitude               DECIMAL(9,7) NULL,
local_authority_id      SMALLINT UNSIGNED NOT NULL,
business_type_id       INT UNSIGNED NOT NULL,
scheme_id              TINYINT UNSIGNED NOT NULL,
PRIMARY KEY (fhrs_id),
KEY idx_establishment_authority (local_authority_id),
KEY idx_establishment_business_type (business_type_id),
KEY idx_establishment_scheme (scheme_id),
KEY idx_establishment_post_code (post_code),
CONSTRAINT fk_establishment_authority
  FOREIGN KEY (local_authority_id) REFERENCES local_authority (local_authority_id)
  ON DELETE RESTRICT ON UPDATE CASCADE,
CONSTRAINT fk_establishment_business_type
  FOREIGN KEY (business_type_id) REFERENCES business_type (business_type_id)
  ON DELETE RESTRICT ON UPDATE CASCADE,
CONSTRAINT fk_establishment_scheme
  FOREIGN KEY (scheme_id) REFERENCES scheme (scheme_id)
  ON DELETE RESTRICT ON UPDATE CASCADE,
CONSTRAINT chk_longitude CHECK (longitude IS NULL OR longitude BETWEEN -180 AND 180),
CONSTRAINT chk_latitude CHECK (latitude IS NULL OR latitude BETWEEN -90 AND 90)
) ENGINE=InnoDB;

CREATE TABLE inspection (
  inspection_id          INT UNSIGNED NOT NULL AUTO_INCREMENT,
  establishment_id       INT UNSIGNED NOT NULL,
  rating_id             SMALLINT UNSIGNED NOT NULL,
  rating_date           DATE NULL,
  hygiene_score         TINYINT UNSIGNED NULL,
  structural_score      TINYINT UNSIGNED NULL,
  confidence_in_management_score TINYINT UNSIGNED NULL,
  new_rating_pending    TINYINT(1) NOT NULL DEFAULT 0,
  PRIMARY KEY (inspection_id),
  UNIQUE KEY uq_inspection_establishment (establishment_id),
  KEY idx_inspection_rating (rating_id),
  CONSTRAINT fk_inspection_establishment
    FOREIGN KEY (establishment_id) REFERENCES establishment (fhrs_id)
    ON DELETE CASCADE ON UPDATE CASCADE,
  CONSTRAINT fk_inspection_rating
    FOREIGN KEY (rating_id) REFERENCES rating_value (rating_id)
    ON DELETE RESTRICT ON UPDATE CASCADE,
  CONSTRAINT chk_hygiene CHECK (hygiene_score IS NULL OR hygiene_score BETWEEN 0 AND 30),
  CONSTRAINT chk_structural CHECK (structural_score IS NULL OR structural_score BETWEEN 0 AND 30),
  CONSTRAINT chk_confidence CHECK (confidence_in_management_score IS NULL OR confidence_in_management_score BETWEEN 0 AND 30),
  CONSTRAINT chk_rating_date CHECK (rating_date IS NULL OR rating_date >= '2000-01-01'),
  CONSTRAINT chk_new_rating_pending CHECK (new_rating_pending IN (0, 1))
) ENGINE=InnoDB;

CREATE USER IF NOT EXISTS 'fhrs_app'@'localhost' IDENTIFIED BY 'change_me_app';
GRANT SELECT ON fhrs.* TO 'fhrs_app'@'localhost';

CREATE USER IF NOT EXISTS 'fhrs_build'@'localhost' IDENTIFIED BY 'change_me_build';
GRANT SELECT, INSERT, UPDATE, DELETE ON fhrs.* TO 'fhrs_build'@'localhost';

FLUSH PRIVILEGES;

```

Appendix B. SQL commands for the research questions

These SQL commands answer the six research questions and validate the loaded one-to-one inspection condition. Example filter values are retained for reproducibility. The second query for question 1 and the second query for question 5 use `business_type_id = 1`, which represents Restaurant, Cafe and Canteen, the most common type overall.

```
USE fhirs;
-- Validation: every loaded establishment should have one inspection
-- Expected result: 0
SELECT COUNT(*) AS establishments_without_inspection
FROM establishment e
LEFT JOIN inspection i ON i.establishment_id = e.fhirs_id
WHERE i.inspection_id IS NULL;
```

```
SELECT la.name AS authority,
       COUNT(*) AS rated_establishments,
       SUM(rv.value_code = '5') AS rated_5,
       ROUND(100 * SUM(rv.value_code = '5') / COUNT(*), 1) AS pct_rated_5
FROM establishment e
JOIN inspection i ON i.establishment_id = e.fhirs_id
JOIN rating_value rv ON rv.rating_id = i.rating_id
JOIN local_authority la ON la.local_authority_id = e.local_authority_id
WHERE rv.is_numeric = 1
GROUP BY la.local_authority_id, la.name
ORDER BY pct_rated_5 DESC;
```

```
SELECT la.name AS authority,
       bt.description AS business_type,
       COUNT(*) AS rated_establishments,
       ROUND(100 * SUM(rv.value_code = '5') / COUNT(*), 1) AS pct_rated_5
FROM establishment e
JOIN inspection i ON i.establishment_id = e.fhirs_id
JOIN rating_value rv ON rv.rating_id = i.rating_id
JOIN local_authority la ON la.local_authority_id = e.local_authority_id
JOIN business_type bt ON bt.business_type_id = e.business_type_id
WHERE rv.is_numeric = 1 AND e.business_type_id = 1
GROUP BY la.local_authority_id, la.name, bt.description
ORDER BY pct_rated_5 DESC;
```

```
SELECT e.business_name,
       e.post_code,
       rv.value_code AS rating,
       i.hygiene_score,
       i.structural_score,
       i.confidence_in_management_score
FROM establishment e
JOIN inspection i ON i.establishment_id = e.fhirs_id
JOIN rating_value rv ON rv.rating_id = i.rating_id
WHERE rv.value_code IN ('0', '1', '2')
      AND e.post_code LIKE 'BN1%'
ORDER BY rv.value_code, e.business_name;
```

```
SELECT bt.description AS business_type,
       COUNT(*) AS scored_establishments,
       ROUND(AVG(i.hygiene_score), 1) AS mean_hygiene,
       ROUND(AVG(i.structural_score), 1) AS mean_structural,
       ROUND(AVG(i.confidence_in_management_score), 1) AS mean_confidence
FROM establishment e
JOIN inspection i ON i.establishment_id = e.fhirs_id
JOIN business_type bt ON bt.business_type_id = e.business_type_id
```

```

WHERE i.hygiene_score IS NOT NULL
GROUP BY bt.business_type_id, bt.description
HAVING COUNT(*) >= 20
ORDER BY mean_hygiene DESC;

```

```

SELECT la.name AS authority,
       COUNT(*) AS total_establishments,
       SUM(rv.value_code = 'AwaitingInspection') AS awaiting_inspection,
       SUM(i.new_rating_pending = 1) AS new_rating_pending,
       ROUND(100 * SUM(rv.value_code = 'AwaitingInspection' OR i.new_rating_pending = 1) / COUNT(*), 1)
         AS pct_awaiting_or_pending
FROM establishment e
JOIN inspection i ON i.establishment_id = e.fhrs_id
JOIN rating_value rv ON rv.rating_id = i.rating_id
JOIN local_authority la ON la.local_authority_id = e.local_authority_id
WHERE la.local_authority_id = 402
GROUP BY la.local_authority_id, la.name;

```

```

SELECT counts.authority,
       counts.business_type,
       counts.n AS establishments
FROM (
  SELECT la.local_authority_id,
         la.name AS authority,
         bt.description AS business_type,
         COUNT(*) AS n
  FROM establishment e
  JOIN local_authority la ON la.local_authority_id = e.local_authority_id
  JOIN business_type bt ON bt.business_type_id = e.business_type_id
  GROUP BY la.local_authority_id, la.name, bt.business_type_id, bt.description
) AS counts
JOIN (
  SELECT local_authority_id, MAX(n) AS max_n
  FROM (
    SELECT la.local_authority_id, e.business_type_id, COUNT(*) AS n
    FROM establishment e
    JOIN local_authority la ON la.local_authority_id = e.local_authority_id
    GROUP BY la.local_authority_id, e.business_type_id
  ) AS per_type
  GROUP BY local_authority_id
) AS top ON top.local_authority_id = counts.local_authority_id AND top.max_n = counts.n
ORDER BY counts.authority;

```

```

SELECT la.name AS authority,
       bt.description AS business_type,
       COUNT(*) AS rated_establishments,
       ROUND(AVG(CAST(rv.value_code AS UNSIGNED)), 2) AS mean_rating
FROM establishment e
JOIN inspection i ON i.establishment_id = e.fhrs_id
JOIN rating_value rv ON rv.rating_id = i.rating_id
JOIN local_authority la ON la.local_authority_id = e.local_authority_id
JOIN business_type bt ON bt.business_type_id = e.business_type_id
WHERE rv.is_numeric = 1 AND e.business_type_id = 1
GROUP BY la.local_authority_id, la.name, bt.description
ORDER BY mean_rating ASC;

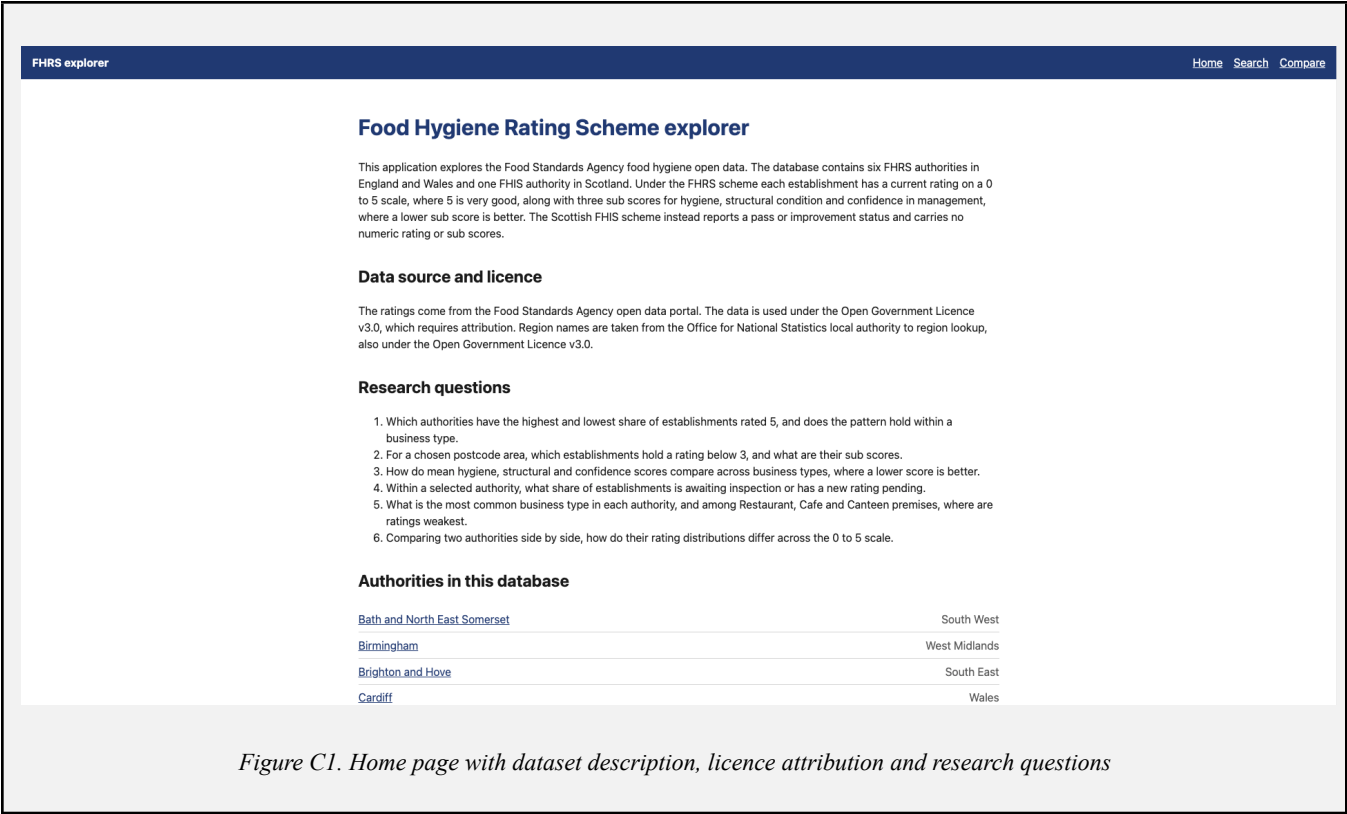
```

```

SELECT rv.value_code AS rating,
       SUM(e.local_authority_id = 402) AS birmingham,
       SUM(e.local_authority_id = 556) AS cardiff
FROM establishment e
JOIN inspection i ON i.establishment_id = e.fhrs_id
JOIN rating_value rv ON rv.rating_id = i.rating_id
WHERE e.local_authority_id IN (402, 556) AND rv.is_numeric = 1
GROUP BY rv.value_code, rv.sort_order
ORDER BY rv.sort_order;

```

Appendix C. Application screenshots



Search establishments

Authority	Business type	Postcode area	Minimum rating
All authorities	All business types	BN1	Any
Maximum rating			
2 or lower			
Apply filters			

Showing page 1, up to 25 results per page.

Establishments matching the current filters

Name	Authority	Business type	Postcode	Rating
Asiamart And Cute Tea	Brighton and Hove	Retailers - other	BN1 4JA	Rated 1 (major improvement necessary)
Bandstand Cafe	Brighton and Hove	Restaurant/Cafe/Canteen	BN1 2LN	Rated 1 (major improvement necessary)
Betterday	Brighton and Hove	Restaurant/Cafe/Canteen	BN1 1AB	Rated 2 (improvement necessary)
Blakes Newsagents And Post Office	Brighton and Hove	Retailers - other	BN1 7GE	Rated 2 (improvement necessary)
Brighton Pizza And Chicken	Brighton and Hove	Restaurant/Cafe/Canteen	BN1 4JD	Rated 1 (major improvement necessary)
Carlito Burrito	Brighton and Hove	Restaurant/Cafe/Canteen	BN1 4GU	Rated 2 (improvement necessary)
Fortune Supermarket	Brighton and Hove	Retailers - other	BN1 2HP	Rated 1 (major improvement necessary)
Jolliffes Coffee Shop	Brighton and Hove	Restaurant/Cafe/Canteen	BN1 4AL	Rated 2 (improvement necessary)

Figure C2. Search results for BN1 establishments with a maximum rating of 2

[Back to home](#)

Birmingham

Region: West Midlands | [Authority website](#)

Summary

Headline figures for Birmingham

Establishments	10037
With a numeric rating	8167
Rated 5	5380 (66 per cent of rated)
Awaiting inspection	1620
New rating pending	124
Mean hygiene score	4.2
Mean structural score	4.8
Mean confidence score	5.1

A lower mean sub score is better.

Rating distribution

The table gives the count for each rating. The bar beside each count shows the same figure as a proportion of the largest bar.

Distribution of ratings for Birmingham

Rating	Count	Share
Rated 0 (urgent improvement necessary)	83	
Rated 1 (major improvement necessary)	387	
Rated 2 (improvement necessary)	316	
Rated 3 (generally satisfactory)	694	
Rated 4 (good)	1307	
Rated 5 (very good)	5380	
Awaiting inspection	1620	

Figure C3. Birmingham authority summary with rating distribution and mean sub scores

Shetland Islands

Region: Scotland | [Authority website](#)

Summary

Headline figures for Shetland Islands

Establishments	278
With a numeric rating	0
Rated 5	0
Awaiting inspection	29
New rating pending	0
Mean hygiene score	n/a
Mean structural score	n/a
Mean confidence score	n/a

A lower mean sub score is better.

Rating distribution

The table gives the count for each rating. The bar beside each count shows the same figure as a proportion of the largest bar.

Distribution of ratings for Shetland Islands

Rating	Count	Share
Pass	144	
Pass and eat safe	2	
Improvement required	29	
Awaiting inspection	29	
Awaiting publication	0	
Exempt	74	

[See establishments in Shetland Islands](#)

Maples, Scotland, Scotland, Scotland, Scotland, Scotland

Figure C4. Shetland Islands authority summary showing FHIS rating values

[Back to search](#)

200 Degrees Coffee Shop

FHRSID 979940, council reference 755909

Current rating

Rated 5 (very good) given on 2025-09-16

Sub scores

A lower sub score is better. The overall score is the three sub scores added together.

Sub scores for 200 Degrees Coffee Shop

Hygiene	5
Structural	5
Confidence in management	5
Overall (derived)	15

Address

115 Queen Street
Cathays
Cardiff
CF10 2BH

Details

Establishment details

Authority	Cardiff
Region	Wales
Business type	Restaurant/Cafe/Canteen
Scheme	FHRS
Longitude	-3.1731083

Figure C5. Establishment detail page with sub scores, overall score and geocode

Compare two authorities

Choose two authorities to see their rating distributions side by side across the numeric 0 to 5 scale.

First authority	Second authority	
Birmingham	Cardiff	Compare

Birmingham and Cardiff

Numeric rating counts for Birmingham and Cardiff

Rating	Birmingham	Cardiff
Rated 0	83	8
Rated 1	387	49
Rated 2	316	44
Rated 3	694	258
Rated 4	1307	557
Rated 5	5380	2117

Data source: Food Standards Agency, Food Hygiene Rating Scheme open data, licensed under the Open Government Licence v3.0. Region names from the Office for National Statistics, licensed under the Open Government Licence v3.0.

Figure C6. Birmingham and Cardiff comparison page

localhost:3000/api/authorities/857/summary

pretty-print

```
{
  "authority": {
    "local_authority_id": 857,
    "name": "Bath and North East Somerset",
    "website": "https://www.bathnes.gov.uk/services/environment/food-safety",
    "email": "public.protection@bathnes.gov.uk",
    "region": "South West"
  },
  "summary": {
    "total": 1841,
    "rated": "1731",
    "rated_5": "1254",
    "awaiting_inspection": "7",
    "new_rating_pending": "10",
    "mean_hygiene": "5.0",
    "mean_structural": "5.4",
    "mean_confidence": "5.1"
  },
  "distribution": [
    {
      "value_code": "0",
      "display_text": "Rated 0 (urgent improvement necessary)",
      "is_numeric": 1,
      "sort_order": 0,
      "n": 1
    },
    {
      "value_code": "1",
      "display_text": "Rated 1 (major improvement necessary)",
      "is_numeric": 1,
      "sort_order": 1,
      "n": 13
    },
    {
      "value_code": "2",
      "display_text": "Rated 2 (improvement necessary)",
      "is_numeric": 1,
      "sort_order": 2,
      "n": 29
    },
    {
      "value_code": "3",
      "display_text": "Rated 3 (generally satisfactory)",
      "is_numeric": 1,
      "sort_order": 3,
      "n": 166
    },
    {
      "value_code": "4",
      "display_text": "Rated 4 (good)",
      "is_numeric": 1,
      "sort_order": 4,
      "n": 268
    },
    {
      "value_code": "5",
      "display_text": "Rated 5 (very good)",
      "is_numeric": 1,
      "sort_order": 5,
      "n": 1254
    },
    {
      "value_code": "AwaitingInspection",
      "display_text": "Awaiting inspection",
      "is_numeric": 0,
      "sort_order": 6,
      "n": 7
    }
  ]
}
```

Figure C7. Read-only JSON authority summary endpoint