

```
chain_of(
  with_elements(load_postgres_table(("public", "patient"), ["id"], [Int32])),
  flatten())
```



The diagram illustrates a complex computational graph, likely representing a neural network architecture or a data processing pipeline. The graph is composed of several interconnected nodes and edges, with a large blue shaded region and a red shaded region highlighting specific areas.

Key Components and Flow:

- Inputs and Initial Processing:** The graph starts with inputs from the left, including a green circle, a yellow box labeled "head", and a green box labeled "load_table('patient' ['id']) SELECT id FROM patient". These feed into a yellow box labeled "1".
- Blue Shaded Region:** This region contains a large green box labeled "load_table('patient' ['mn'], ['id']) SELECT mn FROM patient WHERE id = ?". It also includes a yellow box labeled "head" and a yellow box labeled "1".
- Red Shaded Region:** This region contains a green box labeled "cardinality(x1to1)".
- Output and Further Processing:** The graph continues with a yellow box labeled "head", a green box labeled "output()", a yellow box labeled "1", and a green box labeled "column(1)". These feed into a yellow box labeled "1".
- Right Side Nodes:** The right side of the image shows a series of nodes and edges, including "BlockOf x1to1", "EntityShape DATABASE", "TupleOf", "EntityShape 'patient'", "Int32", "String", and "BlockOf x0toN". These nodes are connected by dotted lines, indicating a flow or relationship between them.

The graph uses various colors and shapes to represent different types of nodes and operations, with arrows indicating the direction of data flow. The blue and red shaded regions likely represent different layers or components of the network.







```
chain_of(with_elements(load_postgres_table(("public", "patient"), ["id"], [Int32])),
  flatten(),
  with_elements(
    chain_of(
      load_postgres_table(("public", "patient"), ["mrn"], [String], ["id"]),
      block_cardinality(x1to1))),
  flatten(),
  with_elements(
    chain_of(
      output(),
      column(1))))
```















