The ==project (π), select (σ), join (⋈)== etc symbols can be easily added via the unicode values:

Symbol	Unicode
σ	03c3
π	03c0
⋈	22c8

Subscript x^2

Suppose we have the following 4 relations:

HOTEL (hotel_no, hotel_name, hotel_city)
ROOM (room_no, hotel_no, room_type, room_price)
BOOKING (hotel_no, guest_no, bdate_from, bdate_to, room_no)

GUEST (guest_no, guest_name, guest_address)

1. List the numbers, names and cities of all hotels

- All hotel_no, hotel_name, hotel_city attributes are in hotel, no conditions.
- $R = \pi_{hotel\_no,\ hotel\_name,\ hotel\_city} HOTEL$

2. List all single rooms with a price below $50

- $R = \sigma_{room\_type = 'single'\ and\ room\_price<50} ROOM$

3. List the numbers and names of all hotels in Melbourne.

- $R = \pi_{hotel\_no,\ hotel\_name} (\sigma_{hotel\_city= 'Melbourne'} HOTEL)$

   Or

- $R1 = \sigma_{hotel\_city= 'Melbourne'} HOTEL$
- $R = _{hotel\_no,\ hotel\_name} R1$

4. List all numbers and names of hotels which have a presidential suite room.

- $R = \pi_{...}\sigma_{...} (HOTEL \bowtie ROOM)$
- $R1 = \pi_{hotel\_no,\ hotel\_name} HOTEL$
- $R2 = \pi_{hotel\_no}(\sigma_{room\_type = 'presidential\ suite'} ROOM)$
- $R = R1 \bowtie R2$

5. List the price and type of all rooms at the Grosvenor Hotel GROSVENOR_NO = π_{hotel_no} ($\sigma_{hotel\_name = 'Grosvenor'}$ HOTEL)

R = $\pi_{room\_price, room\_type}$ (GROSVENOR_NO $\bowtie$ ($\pi_{hotel\_no, room\_price, room\_type}$ ROOM)) or

R = $\pi_{room\_price, room\_type}$ ((π_{hotel_no} ($\sigma_{hotel\_name = 'Grosvenor'}$ HOTEL)) $\bowtie$ ($\pi_{hotel\_no, room\_price, room\_type}$ ROOM))

6. List all numbers, names and addresses of guests currently staying in deluxe room of any hotel (assume that if the guest has a tuple in the BOOKING relation, then they are currently staying in the hotel)

DELUXE_NO = $\pi_{hotel\_no, room\_no}$ ($\sigma_{room\_type = 'deluxe'}$ ROOM)
GUEST_DELUXE_NO = π_{guest_no} (DELUXE_NO $\bowtie$ ($\pi_{guest\_no, hotel\_no, room\_no}$ BOOKING))

R = GUEST_DELUXE_NO $\bowtie$ GUEST
or
R = π_{guest_no} (($\pi_{hotel\_no, room\_no}$ ($\sigma_{room\_type = 'deluxe'}$ ROOM)) $\bowtie$

($\pi_{guest\_no, hotel\_no, room\_no}$ BOOKING))) $\bowtie$ GUEST

7. List all numbers, names and addresses of guests currently staying at the Grosvenor Hotel (assume that if the guest has a tuple in the BOOKING relation, then they are currently staying in the hotel)

GROSVENOR_NO = π_{hotel_no} ($\sigma_{hotel\_name = 'Grosvenor'}$ HOTEL)

GROSVENOR_BOOKING = π_{guest_no} (GROSVENOR_NO $\bowtie$ ($\pi_{guest\_no, hotel\_no}$ BOOKING)) R = GROSVENOR_BOOKING $\bowtie$ GUEST
or
R = π_{guest_no} ((π_{hotel_no} ($\sigma_{hotel\_name = 'Grosvenor'}$ HOTEL)) $\bowtie$

($\pi_{guest\_no, hotel\_no}$ BOOKING))) $\bowtie$ GUEST

4.2.2 Additional Relational Algebra Exercise

CUSTOMER (cust_id, cust_name, cust_address)

PRODUCT (prod_id, prod_desc, prod_unitprice, prod_stock)

STAFF(staff_name, staff_position)

SALE (cust_id, sale_date, prod_id, sale_qty, sold_by)
*Note that sold_by value is the name of staff who made the sale

1. List ids, names of customers and descriptions of products bought by the customers. How many tuples will be returned by the relational algebra query that you have constructed as your answer?

$R1 = \pi_{cust\_id, cust\_name} CUSTOMER \quad R2 = (\pi_{cust\_id, prod\_id} SALE) \bowtie R1$

$R = \pi_{cust\_id, cust\_name, prod\_desc} ((\pi_{prod\_id, prod\_desc} PRODUCT) \bowtie R2)$

$\rightarrow$ 5 tuples

2. List all names which are shared by customers and staff. $R1 = \pi_{cust\_name} CUSTOMER$

$R2 = \pi_{staff\_name} STAFF$

$R = R1 \cap R2$

3. List ids and descriptions of products that haven't been sold

$R = ((\pi_{prod\_id} PRODUCT) - (\pi_{prod\_id} SALE)) \bowtie (\pi_{prod\_id, prod\_desc} PRODUCT)$

4. List names of clerks who don't have any sales yet

$R = (\pi_{staff\_name}(\sigma_{staff\_position="Clerk"} STAFF)) - (\pi_{sold\_by} SALE)$

5. List positions of staff who have made sales

$R = \pi_{staff\_position} ((\pi_{sold\_by} SALE) \bowtie_{(staff\_name=sold\_by)} STAFF)$