**CREATE A DICTIONARY** named insulin\_variables with key:value pairs of insulin\_entry\_code:(MAX\_DEL\_DOSE, DOSE\_INCR, UNITS\_PER\_PEN)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Insulin Pen | Code | MAX\_DEL\_DOSE | DOSE\_INCR | UNITS\_PER\_PEN |
| Lantus SoloStar | 1 | 80 | 1 | 300 |
| Novolog FlexPen | 2 | 60 | 1 | 300 |
| Semglee Pen | 3 | 80 | 1 | 300 |
| Humulin R U-500 KwikPen | 4 | 300 | 5 | 1500 |
| Tresiba U-200 FlexTouch | 5 | 160 | 2 | 600 |

**DISPLAY WELCOME MESSAGE**

**GET INPUT**: request selection of insulin product by presenting a table of insulins with an assigned entry\_code (Table 1 contains a small sample of insulin pens) or enter SENTINAL to exit. Note that entry\_code should be the same as matched key in dictionary:

|  |  |
| --- | --- |
| Insulin Pen | Code |
| Lantus SoloStar | 1 |
| Novolog FlexPen | 2 |
| Semglee Pen | 3 |
| Humulin R U-500 KwikPen | 4 |
| Tresiba U-200 FlexTouch | 5 |

entry\_code = int(input())

**VALIDATE INPUT**: ensure entry\_code is integer between 1 – last insulin code. Request re-entry if needed.

**DECLARE CONSTANTS**: use user input to access dictionary values. Values in dictionary are a tuple so unpack tuple to create constants:

MAX\_DEL\_DOSE, DOSE\_INCR, UNITS\_PER\_PEN = insulin\_variables[entry\_code]

**GET INPUT:** user enters number of doses per day

doses\_per\_day = int(input(‘Enter the number of doses per day:’))

**GET INPUT:** request number of units per each dose.

**DECLARE VARIABLE:**

dose\_entry = 1

unit\_per\_dose = 0

extra\_needles\_total = 0

total\_units\_daily = 0

**CALCULATE NUMBER OF EXTRA NEEDLES NEEDED WITH WHILE LOOP FOR EACH DOSE**

While dose\_entry <= doses\_per\_day

Display ‘Enter the number of units for dose [dose\_entry]’

unit\_per\_dose = input()

**VALIDATE INPUT:** ensure entry is an integer. Also ensure that entry is evenly divisible by DOSE\_INCR. This is to ensure that the prescribed number of units of insulin are able to be delivered by the device.

extra\_needles\_for\_dose = 0

if unit\_per\_dose <= MAX\_DEL\_DOSE then

extra\_needles\_for\_dose = 0

else

extra\_needles\_for\_dose = ceil(unit\_per\_dose/MAX\_DEL\_DOSE) – 1

extra\_needles\_total += extra\_needles\_for\_dose

dose\_entry += 1

total\_units\_daily += unit\_per\_dose

**CALCULATE NUMBER OF NEEDLES**

pens\_needed = ceil(total\_units\_daily/UNITS\_PER\_PEN)

total\_needles = (doses\_per\_day + extra\_needles\_total)\* days\_supply + (pens\_needed – 1)

**DISPLAY TOTAL**

Display total\_needles