Question 1 (20 pts)

Consider the following English sentences.
a. Black tea is a type of tea and cheese is not.
b. Black tea can blend with Green Tea
c. If an element is a not a tea, then no tea can blend with it.
d. Some tea are unwilted and unoxidized.
e. Some tea can blend with all tea except the unoxidized ones.

1. (5 pts) Convert the sentences into first-order predicate logic. Use the following lexicon:
   tea (X) - X is a type of tea.
   unoxidize (X) - X has NOT been oxidized.
   wilt (X) - X has been wilted.
   blend (X, Y) - X can blend with Y.

   (a) $tea(Black) \wedge \neg tea(Cheese)$

   (b) $blend(tea(Black), tea(Green))$

   (c) $\forall x \forall y \neg tea(x) \Rightarrow \neg blend(tea(x), tea(y))$

   (d) $\exists x tea(x) \wedge \neg wilt(tea(x)) \wedge unoxidize(tea(x))$

   (e) $\exists x \forall y tea(x) \wedge \neg unoxidize(tea(x)) \Rightarrow blend(tea(y), tea(x))$

2. (10 pts) Follow the steps on text book to convert the logic statements into CNF.

   Steps: We eliminate implications, then we move the negation inwards,then we standardize the variables,then we perform skolemization, then drop the universal quantifiers and finally distribute 'or' over 'and'.

   (a) $tea(Black) \wedge \neg tea(Cheese)$

   (b) $blend(tea(Black), tea(Green))$

   (c) $tea(x) \vee \neg blend(tea(x), tea(y))$

   (d) $tea(x) \wedge \neg wilt(tea(x)) \wedge unoxydize(tea(x))$:
   $tea(C1) \wedge \neg wilt(tea(C1)) \wedge unoxydize(tea(C1))$

   (e) $\neg tea(x) \vee unoxydize(tea(x)) \wedge blend(tea(y), tea(x))$:
   $\neg tea(C2) \vee unoxydize(tea(C2)) \vee blend(tea(k), tea(C2))$

3. (5 pts) Using FOPL resolution prove the conclusion that Black tea cannot blend with cheese. Number your clauses, and indicate explicitly step-by-step what resolves together, and under what substitution.

   Prove that: $\neg blend(tea(black), \neg tea(cheese))$
   Let us show that the opposite $blend(tea(black), \neg tea(cheese))$ is false.
   Resolution steps:

   (a) $blend(tea(black), \neg tea(cheese))$
   We resolve the above statements by first resolving it with Statement (c), $tea(x) \vee \neg blend(tea(x), tea(y))$ where x = cheese and y = black.

   (b) $tea(cheese)$
   We know from statement (a) that $tea(Black) \wedge \neg tea(Cheese)$ and both black tea and cheese are independent entities.
   Thus we resolve the above statement with $\neg tea(cheese)$

   (c) ϕ after resolution

Question 2 (15 pts)

Use the lexicon:
s – A customer can shop at Costco
o – A Costco customer
i – A customer has a membership identification with him/her
d – Customers will get membership identifications
g – A customer has gold membership
si – A customer has silver membership
pre – A customer can enjoy premium discount
l – A customer can collect loyalty points
c – Carol is a customer

1. All Costco customers can collect loyalty points.
2. All Costco customers can get membership identifications.
3. Costco customers can shop at Costco stores only when they bring membership identifications with them.
4. Carol cannot shop at Costco stores.
5. Costco customers has either gold membership or Silver membership.
6. Carol is a Costco customer.
7. Carol has gold membership but doesn't get premium discount and cannot shop at Costco stores since she forgot to take her Costco identification.
8. Customers with gold memberships can have premium discount only if they have their identification with them while customers with silver memberships cannot.

1. (10 pts) Use propositional logic to determine if the specification is consistent. If the sentences are not consistent, use resolution to derive a contradiction. If they are consistent, use the truth-table method to show at least one model.
   (a) $o \Rightarrow l : \neg o \vee l$
   (b) $o \Rightarrow d : \neg o \vee d$
   (c) $o \wedge i \Rightarrow s : \neg o \vee \neg i \vee s$
   (d) $c \wedge \neg s$
   (e) $o \wedge (g \vee si)$
   (f) $c \wedge o$
   (g) $(c \wedge g \wedge \neg pre) \wedge (\neg i \Rightarrow \neg s) : (c \wedge g \wedge \neg pre) \wedge (i \vee \neg s)$
   (h) $(g \wedge i \Rightarrow pre) \wedge (si \Rightarrow \neg pre) : (\neg g \vee \neg i \vee pre) \wedge (\neg si \vee \neg pre)$

Assigning T/F to variables:

Variables	o	l	d	i	s	c	g	si	pre	
Assigned vals	T	T	T	F	F	T	T	F	F	
Statement 1	$F \lor T$									T
Statement 2	$F \lor T$									T
Statement 3	$F \lor T \lor F$									T
Statement 4	$T \land T$									T
Statement 5	$T \land (T \lor F)$									T
Statement 6	$T \land T$									T
Statement 7	$(T \land T \land T) \land (F \lor T)$									T
Statement 8	$(F \lor T \lor F) \land (T \lor T)$									T

2. (5 pts) What if anything changes if Carol can shop at Costco stores and does the consistency change? Be specific in showing what's different.
   Statement 4 becomes: $c \land s$ and we assign T to S instead of F

Assigning T/F to variables:

Variables	o	l	d	i	s	c	g	si	pre	
Assigned vals	T	T	T	T	T	T	T	F	F	
Statement 1	$F \lor T$									T
Statement 2	$F \lor T$									T
Statement 3	$F \lor F \lor T$									T
Statement 4	$T \land T$									T
Statement 5	$T \land (T \lor F)$									T
Statement 6	$T \land T$									T
Statement 7	$(T \land T \land T) \land (F \lor T)$									T
Statement 8	$(F \lor F \lor F) \land (T \lor T)$									F

Since statement (h) gives a value of False in the Truth Table, we take the negation of the same to check for correctness.
Statement (h): $(\neg g \lor \neg i \lor pre) \land (\neg si \lor \neg pre)$
Opposite of statement (h) may be represented as $(g \land i \land \neg pre) \lor (si \land \ pre)$
Applying rule (g) - $(c \land g \land \neg pre) \land (i \lor \neg s)$ to the above statement, we get
$(c \land g \land \neg pre \lor s) \lor (c \land g \land \neg i \lor s)$
From the list of available statements, it can be seen that no statement exists in the knowledge base that contains the variables present in the new resolved statement.
Initially, we considered statement (h) to be True but upon the discovery that no statement satisfying the above resolution exists, we conclude that the specification is inconsistent

Question 3 (65 pts)

The code has been written in a separate python script along with a README file explaining the code and how it has to be run