Up & Down Game

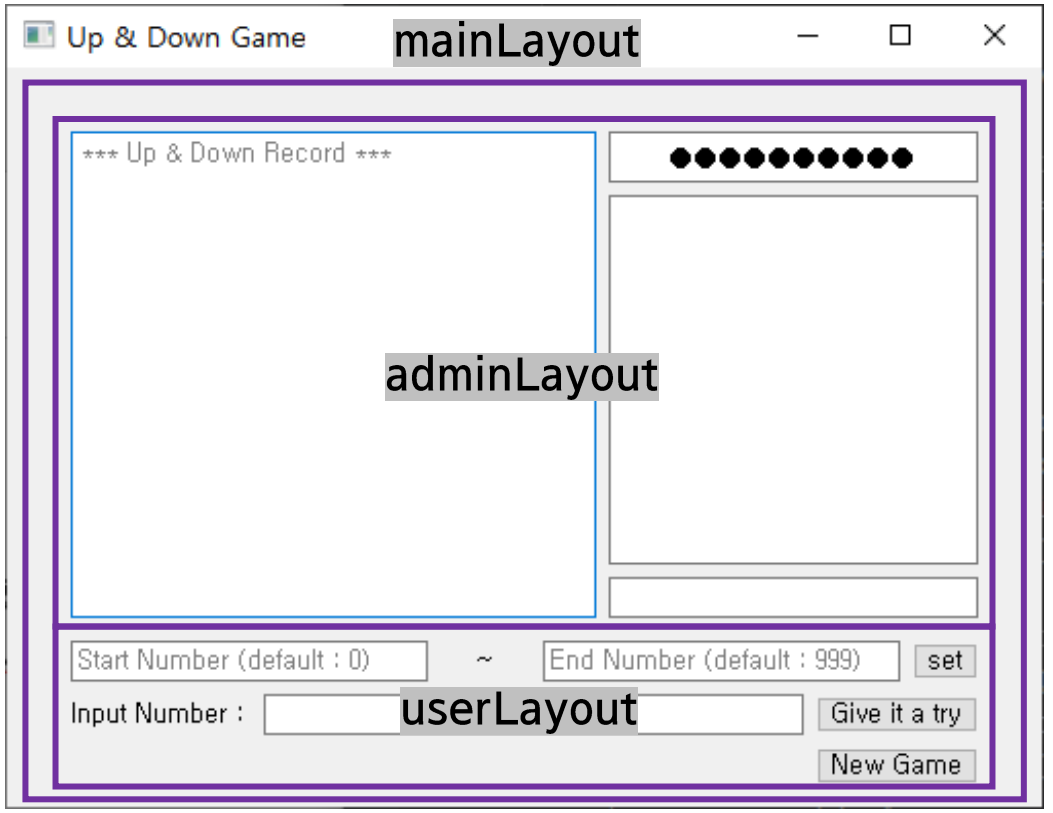
AD Project

20185280 강다윤

UI

game.py

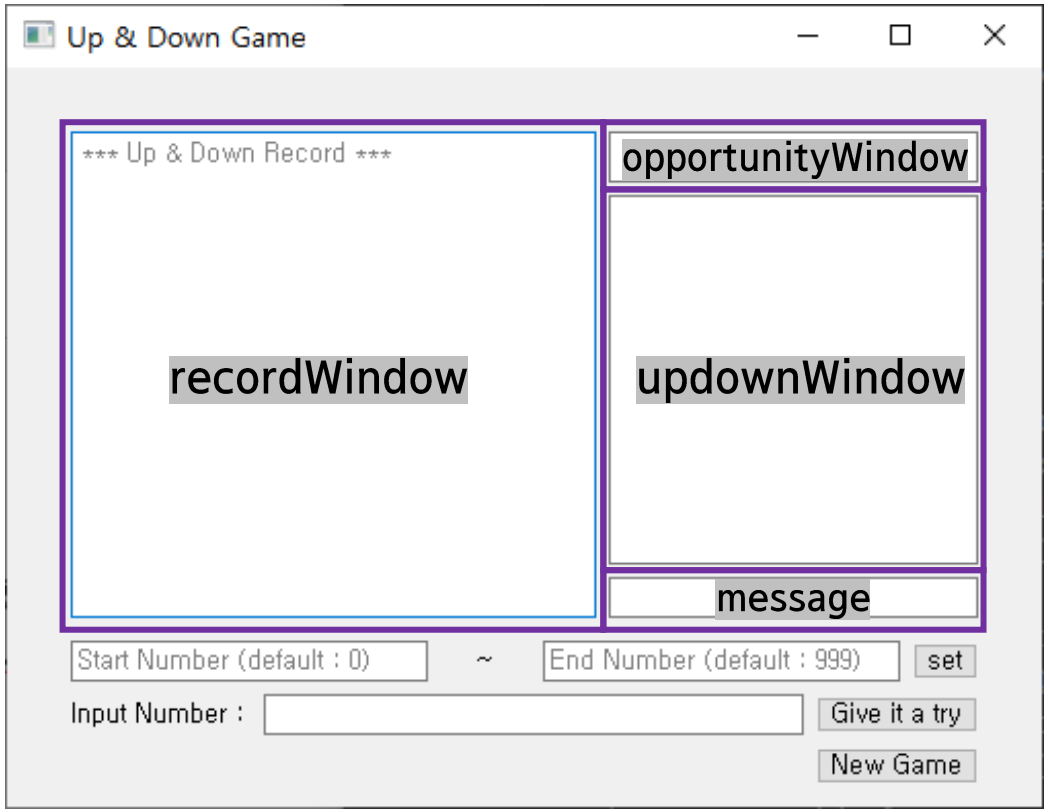
```
# Layout placement
adminLayout = QGridLayout()
adminLayout.addLayout(recordLayout, 0, 0)
adminLayout.addLayout(showLayout, 0, 1)
adminLayout.setContentsMargins(20, 20, 20, 5)

userLayout = QGridLayout()
userLayout.addLayout(rangeLayout, 0, 0)
userLayout.addLayout(inputLayout, 1, 0)
userLayout.setContentsMargins(20, 0, 20, 0)

mainLayout = QGridLayout()
mainLayout.addLayout(adminLayout, 0, 0)
mainLayout.addLayout(userLayout, 1, 0)

self.setLayout(mainLayout)

self.setWindowTitle('Up & Down Game')
```



UI

game.py

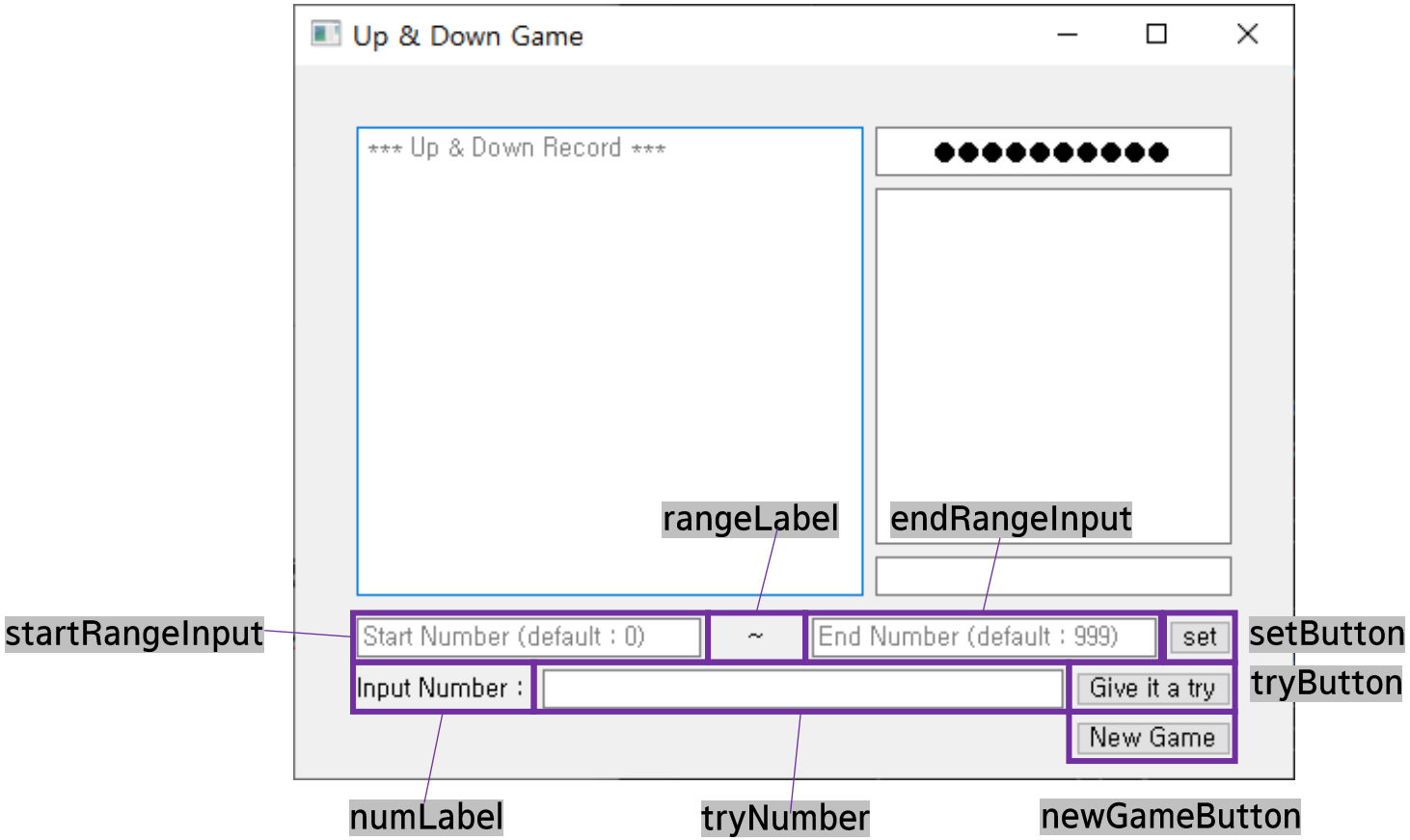
```
# Display widget for recording number with Up and Down
self.recordWindow = QTextEdit()
self.recordWindow.setReadOnly(True)
self.recordWindow.setAlignment(Qt.AlignLeft)
self.recordWindow.setPlaceholderText('*** Up & Down Record ***')

recordLayout = QGridLayout()
recordLayout.addWidget(self.recordWindow, 0, 0)

showLayout = QGridLayout()

# Display widget for remaining opportunity
self.opportunityWindow = QLineEdit()
self.opportunityWindow.setReadOnly(True)
self.opportunityWindow.setAlignment(Qt.AlignCenter)
self.opportunityWindow.setFixedSize(180, 25)
showLayout.addWidget(self.opportunityWindow, 0, 0)

# Display widget for status
self.updownWindow = QTextEdit()
self.updownWindow.setReadOnly(True)
self.updownWindow.setAlignment(Qt.AlignCenter)
self.updownWindow.setFixedSize(180, 180)
showLayout.addWidget(self.updownWindow, 1, 0)

# Display widget for message output
self.message = QLineEdit()
self.message.setReadOnly(True)
self.message.setAlignment(Qt.AlignRight)
showLayout.addWidget(self.message, 2, 0)
```



UI

game.py

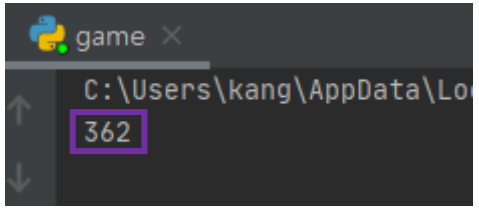
```
# Input widget for setting range, start number
self.startRangeInput = QLineEdit()
self.startRangeInput.setPlaceholderText("Start Number (default : 0)")
rangeLayout.addWidget(self.startRangeInput, 4)

# Display widget '~'
self.rangeLabel = QLabel()
self.rangeLabel.setText("~")
self.rangeLabel.setAlignment(Qt.AlignCenter)
rangeLayout.addWidget(self.rangeLabel, 1)

# Input widget for setting range, end number
self.endRangeInput = QLineEdit()
self.endRangeInput.setPlaceholderText("End Number (default : 999)")
rangeLayout.addWidget(self.endRangeInput, 4)

# Button for setting random number's range
self.setButton = QToolButton()
self.setButton.setText("set")
self.setButton.clicked.connect(self.setClicked)
rangeLayout.addWidget(self.setButton, 3)

# Display widget 'Input Number : '
self.numLabel = QLabel()
self.numLabel.setText("Input Number : ")
inputLayout.addWidget(self.numLabel, 0, 0)

# Input widget for user selected number
self.tryNumber = QLineEdit()
self.tryNumber.setMaxLength(3)
inputLayout.addWidget(self.tryNumber, 0, 1)

# Button for submitting a number
self.tryButton = QToolButton()
self.tryButton.setText('Give it a try')
self.tryButton.clicked.connect(self.tryClicked)
inputLayout.addWidget(self.tryButton, 0, 2)

# Button for a new game
self.newGameButton = QToolButton()
self.newGameButton.setText('New Game')
self.newGameButton.clicked.connect(self.startGame)
inputLayout.addWidget(self.newGameButton, 1, 2)
```



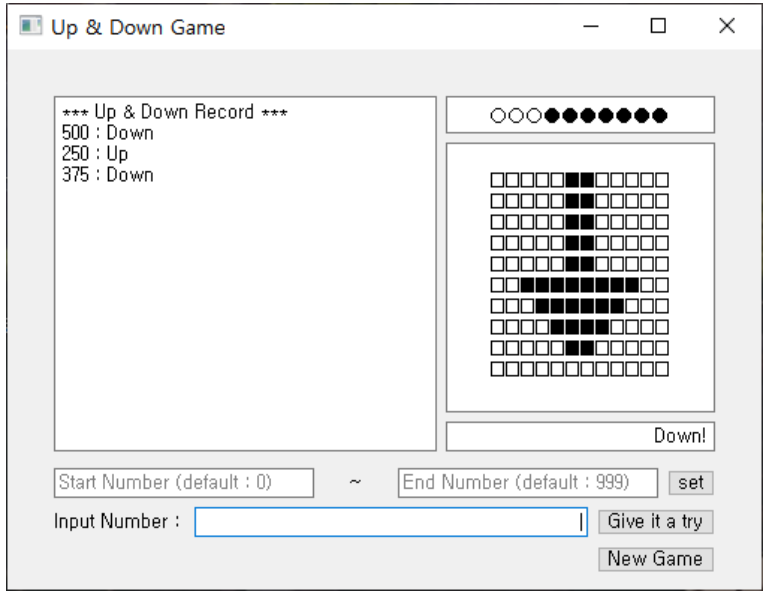
기능 구현

numtry.py

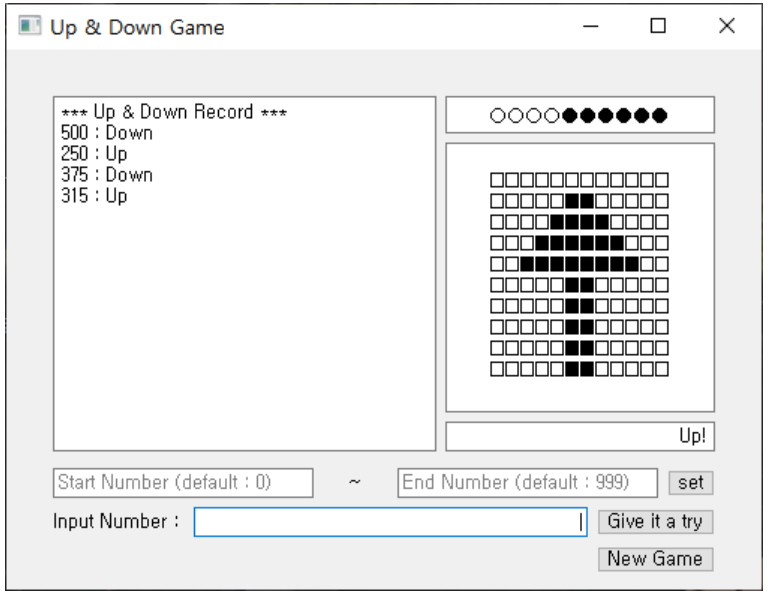
```
print(self.secretNumber)
```



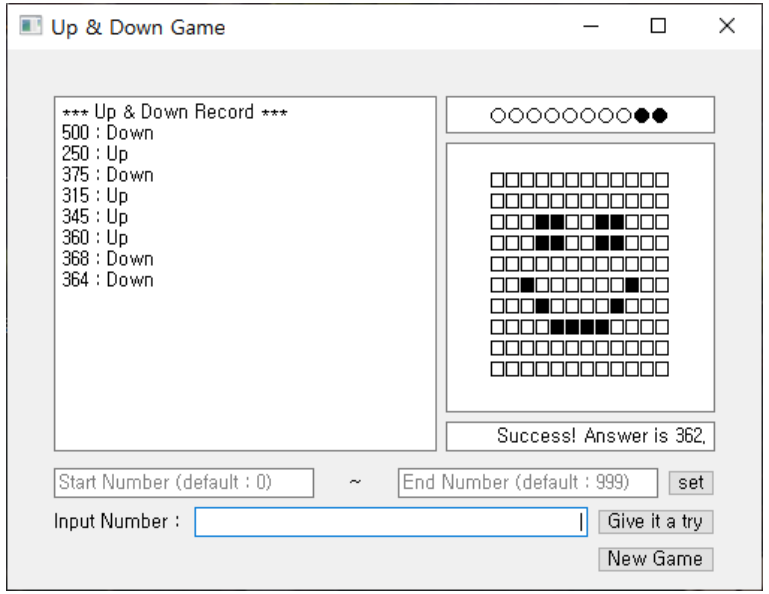
1. 게임 실행 (범위 설정 X)



recordWindow - 375 : Down 추가
opportunityWindow – 기회 하나 감소
updownWindow – Down text 출력
message – Down! 출력



recordWindow – 315 : Up 추가
opportunityWindow – 기회 하나 감소
updownWindow – Up text 출력
message – Up! 출력



updownWindow – success text 출력
message – Success! + 362
(secret number) 출력

기능 구현

1-1. 난수 생성

number.py

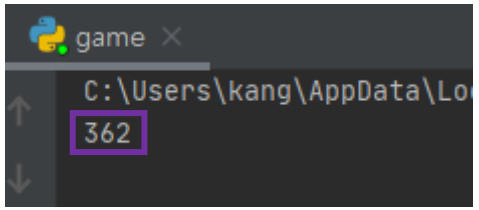
```
def __init__(self):
    self.randomNum = random.randint(0, 999)

def randomNumber(self):
    return random.randint(0, 999)
```

game.py

```
def startGame(self):
    self.updown = Updown()
    self.numtry = Numtry(self.number.randomNum)
    self.gameOver = False
    self.opportunityWindow.setText(self.updown.currentOpportunity())
    self.message.clear()

def newClicked(self):
    self.updown = Updown()
    self.numtry = Numtry(self.number.randomNumber())
    self.gameOver = False
    self.opportunityWindow.setText(self.updown.currentOpportunity())
    self.recordWindow.clear()
    self.updownWindow.clear()
    self.startRangeInput.clear()
    self.endRangeInput.clear()
    self.message.clear()
```

numtry.py

```
print(self.secretNumber)
```



게임 시작 시와
newButton을 눌러 새 게임을 시작 시에
Default 값 0 ~ 999 까지 난수 생성

기능 구현

1-2. 숫자 비교

numtry.py

```
# Record used numbers and Compare input number with secret number
def numtry(self, num):
    self.recordNums.append(num)
    if num != self.secretNumber:
        if num < self.secretNumber:
            return 0
        return 1
    self.isfinished=True

    return 2
```

numtry.py의 numtry 함수에서 숫자 비교

game.py

```
# Case : input number < secret number
if case == 0:
    self.updown.decreaseOpportunity()
    self.message.setText("Up!")
    self.numtry.updownTry(tryNum + " : Up" + "\n")
    self.recordWindow.setText(self.numtry.getRecordUpdown())
    self.updownWindow.setText(self.updown.getUpDisplay())
    self.opportunityWindow.setText(self.updown.currentOpportunity())

# Case : input number > secret number
if case == 1:
    self.updown.decreaseOpportunity()
    self.message.setText("Down!")
    self.numtry.updownTry(tryNum + " : Down" + "\n")
    self.recordWindow.setText(self.numtry.getRecordUpdown())
    self.updownWindow.setText(self.updown.getDownDisplay())
    self.opportunityWindow.setText(self.updown.currentOpportunity())
```

case를 나눠 game.py에서
case에 맞는 window와 message를 설정

기능 구현

1-3. 케이스 별 처리 & message

game.py

```
if self.gameOver == True:
    self.message.setText("Game Over!")
    return

if len(tryNum) > 3 or len(tryNum) < 1:
    self.message.setText("Put number 0 to 999.")
    return

if int(tryNum) in self.numtry.recordNums:
    self.message.setText("You already have it.")
    return

if int(tryNum) not in self.numtry.recordNums:
    self.message.setText("Keep Going.")
```

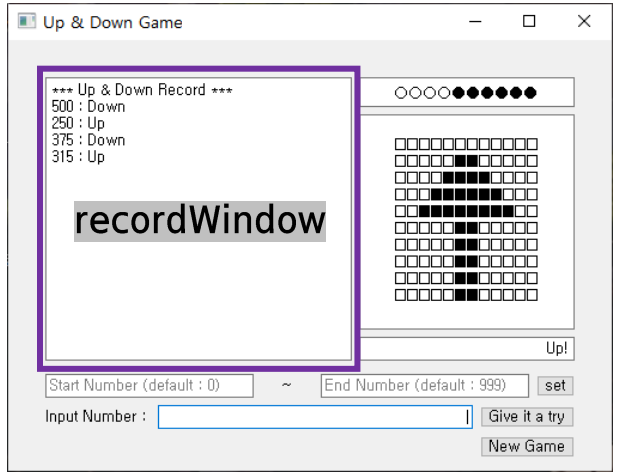
...

```
# Case for success, input number == secret number
if self.numtry.finished():
    self.message.setText("Success! Answer is " + str(self.numtry.secretNumber) + ".")
    self.gameOver = True
    self.updownWindow.setText(self.updown.getSuccessDisplay())

# Case for game over, No opportunity
elif self.updown.getOpportunity() == 0:
    self.message.setText("Fail! Answer was " + str(self.numtry.secretNumber) + ".")
    self.gameOver = True
    self.updownWindow.setText(self.updown.getGameoverDisplay())
```

1. Game over
2. 세 자리 수 이상 또는 입력되지 않았을 때
3. 이미 입력된 숫자를 다시 입력했을 때
4. 새로운 숫자를 입력했을 때
5. 입력된 숫자가 비밀 숫자보다 작을 때
6. 입력된 숫자가 비밀 숫자보다 클 때
7. 성공 했을 때
8. 남은 기회가 없을 때

기능 구현

1-4. window 구현 : recordWindow



numtry.py

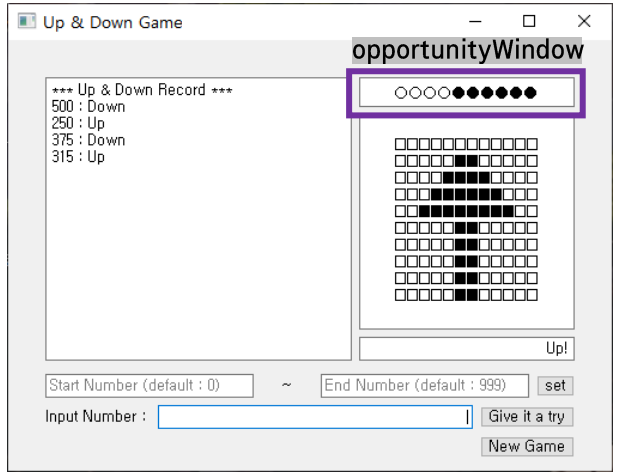
```
# Record used numbers and Compare input number with secret number
def numtry(self, num):
    self.recordNums.append(num)
    if num != self.secretNumber:
        if num < self.secretNumber:
            return 0
        return 1
    self.isfinished=True

    return 2

# Show recording numbers
def getRecordNums(self):
    recordNum = ''
    for i in self.recordNums:
        recordNum += (str(i))
    return recordNum

# Record status, input number with Up and Down
def updownTry(self, updown):
    self.recordUpdowns.append(updown)

# Show recording status
def getRecordUpdown(self):
    recordUpdown = '*** Up & Down Record ***\n'
    for i in self.recordUpdowns:
        recordUpdown += (str(i))
    return recordUpdown
```

game.py

```
# Case : input number < secret number
if case == 0:
    self.updown.decreaseOpportunity()
    self.message.setText("Up!")
    self.numtry.updownTry(tryNum + " : Up" + "\n")
    self.recordWindow.setText(self.numtry.getRecordUpdown())
    self.updownWindow.setText(self.updown.getUpDisplay())
    self.opportunityWindow.setText(self.updown.currentOpportunity())

# Case : input number > secret number
if case == 1:
    self.updown.decreaseOpportunity()
    self.message.setText("Down!")
    self.numtry.updownTry(tryNum + " : Down" + "\n")
    self.recordWindow.setText(self.numtry.getRecordUpdown())
    self.updownWindow.setText(self.updown.getDownDisplay())
    self.opportunityWindow.setText(self.updown.currentOpportunity())
```

기능 구현

1-4. window 구현 : opportunityWindow



updown.py

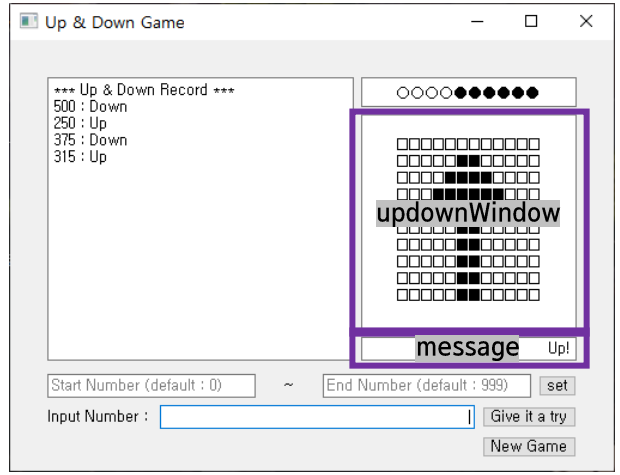
```
text_opportunity = [
    '○○○○○○○○○○',
    '○○○○○○○○○●',
    '○○○○○○○○●●',
    '○○○○○○○●●●',
    '○○○○○○●●●●',
    '○○○○○●●●●●',
    '○○○○●●●●●●',
    '○○○●●●●●●●',
    '○○●●●●●●●●',
    '○●●●●●●●●●',
    '●●●●●●●●●●'
]


def __init__(self):
    self.opportunity = len(self.text_opportunity) - 1


def getOpportunity(self):
    return self.opportunity


def decreaseOpportunity(self):
    self.opportunity -= 1


def currentOpportunity(self):
    return self.text_opportunity[self.opportunity]
```

game.py

```
# Case : input number < secret number
if case == 0:
    self.updown.decreaseOpportunity()
    self.message.setText("Up!")
    self.numtry.updownTry(tryNum + " : Up" + "\n")
    self.recordWindow.setText(self.numtry.getRecordUpdown())
    self.updownWindow.setText(self.updown.getUpDisplay())
    self.opportunityWindow.setText(self.updown.currentOpportunity())

# Case : input number > secret number
if case == 1:
    self.updown.decreaseOpportunity()
    self.message.setText("Down!")
    self.numtry.updownTry(tryNum + " : Down" + "\n")
    self.recordWindow.setText(self.numtry.getRecordUpdown())
    self.updownWindow.setText(self.updown.getDownDisplay())
    self.opportunityWindow.setText(self.updown.currentOpportunity())
```

기능 구현

1-4. window 구현 : updownWindow & message



updown.py

```
text_mainDispaly = [
'''\
  ...
''',
'''\
  ...
''',
'''\
  ...
''',
]

def getUpDisplay(self):
    return self.text_mainDispaly[0]


def getDownDisplay(self):
    return self.text_mainDispaly[1]


def getSuccessDisplay(self):
    return self.text_mainDispaly[2]


def getGameoverDisplay(self):
    return self.text_mainDispaly[3]
```

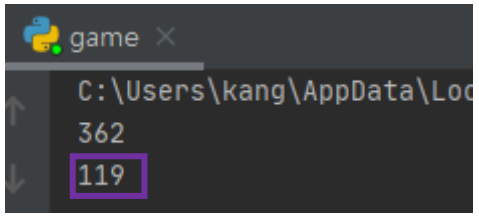
game.py

```
# Case : input number < secret number
if case == 0:
    self.updown.decreaseOpportunity()
    self.message.setText("Up!")
    self.numtry.updownTry(tryNum + " : Up" + "\n")
    self.recordWindow.setText(self.numtry.getRecordUpdown())
    self.updownWindow.setText(self.updown.getUpDisplay())
    self.opportunityWindow.setText(self.updown.currentOpportunity())

# Case : input number > secret number
if case == 1:
    self.updown.decreaseOpportunity()
    self.message.setText("Down!")
    self.numtry.updownTry(tryNum + " : Down" + "\n")
    self.recordWindow.setText(self.numtry.getRecordUpdown())
    self.updownWindow.setText(self.updown.getDownDisplay())
    self.opportunityWindow.setText(self.updown.currentOpportunity())
```

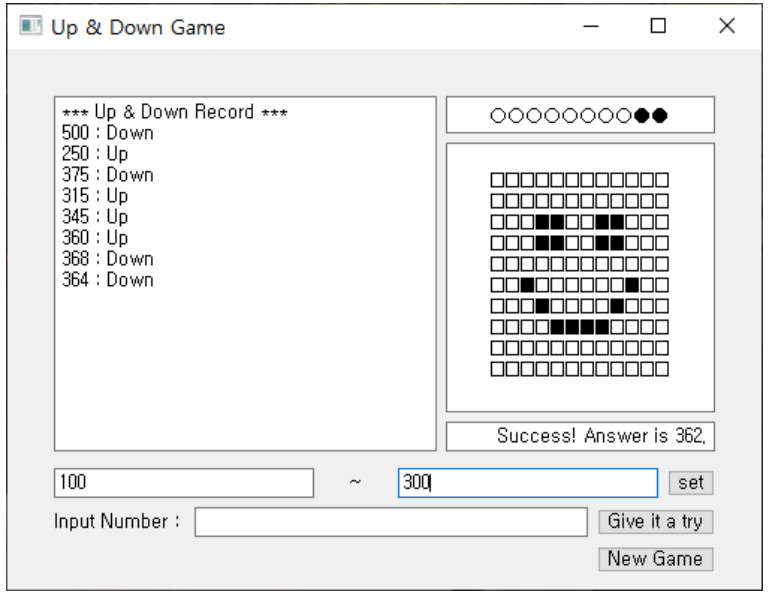
기능 구현

numtry.py

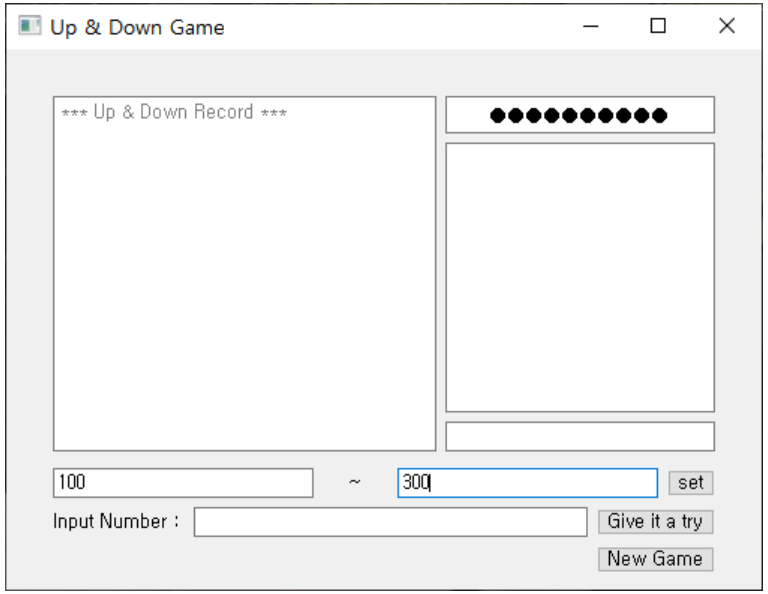
```
print(self.secretNumber)
```



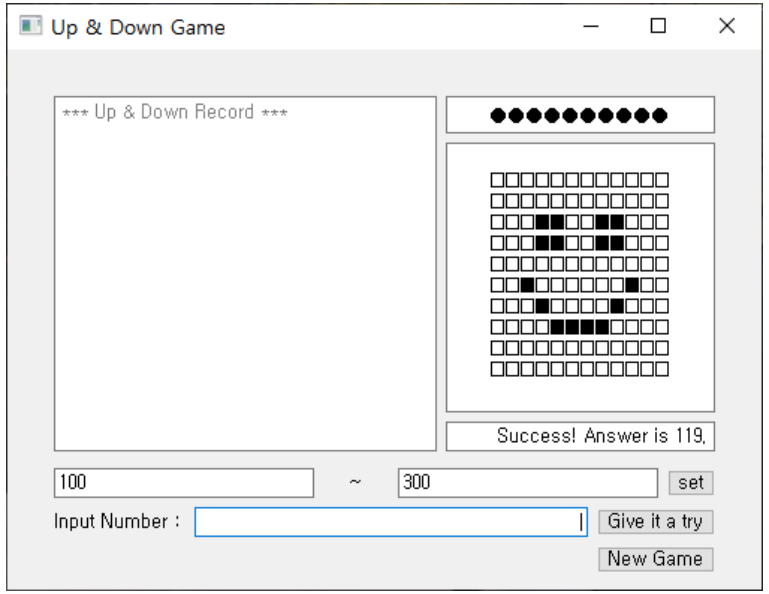
2. 게임 실행 (범위 설정 O)



100 ~ 300 까지 범위 설정
setButton 클릭



recordWindow – clear
opportunityWindow – clear
updownWindow – clear
message – clear
설정 범위 내 난수 생성



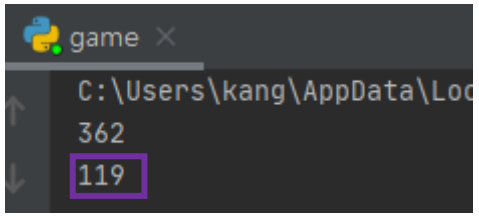
119 (secret number) 입력 시,
Success

기능 구현

2-1. 설정 범위 내 난수 생성 & setClicked

numtry.py

```
print(self.secretNumber)
```



number.py

```
# Set random number in given range
def rangeNumber(self, start, end):
    if start <= end:
        return random.randint(int(start), int(end))
    return
```

game.py

```
def setClicked(self):
    self.updown = Updown()
    self.startInput = int(self.startRangeInput.text())
    self.endInput = int(self.endRangeInput.text())
    self.numtry = Numtry(self.number.rangeNumber(self.startInput, self.endInput))
    self.gameOver = False
    self.opportunityWindow.setText(self.updown.currentOpportunity())
    self.recordWindow.clear()
    self.updownWindow.clear()
    self.message.clear()

    # Case : User input wrong range, start number > end number
    if self.startInput > self.endInput:
        self.message.setText("Input right range.")
        return
```

number.py에서 rangeNumber 함수 생성
start가 end보다 작거나 같을 경우에만 값을 return

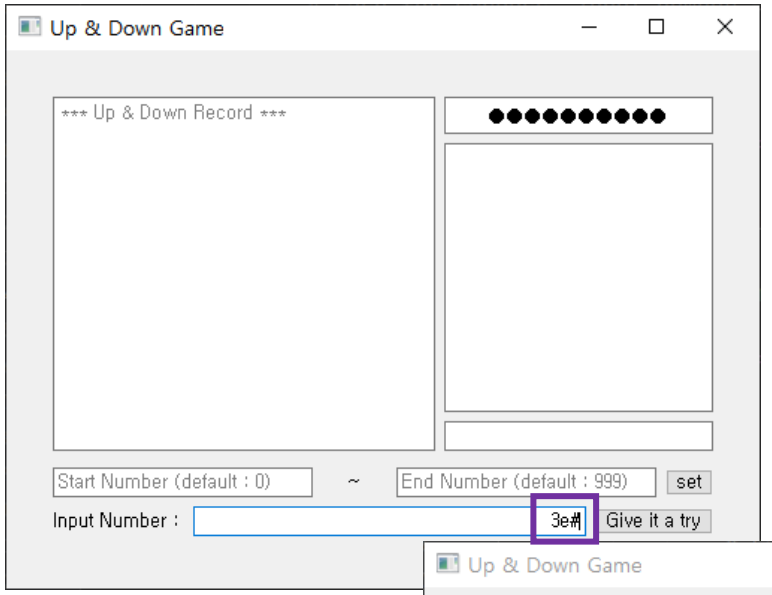
self.startInput 과 self.endInput을 함수의 인자로 넣어줌

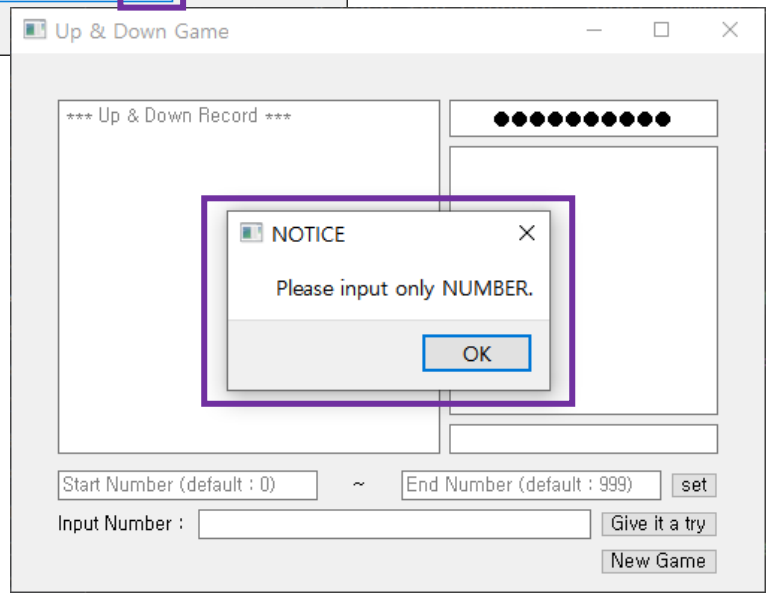
startInput이 endInput보다 더 큰 경우에는 메시지 출력

setButton 누르면 모든 창이 초기화 되도록 설정

예외 처리

1. 입력된 값이 십진 정수가 아닐 때 – Input Number





game.py

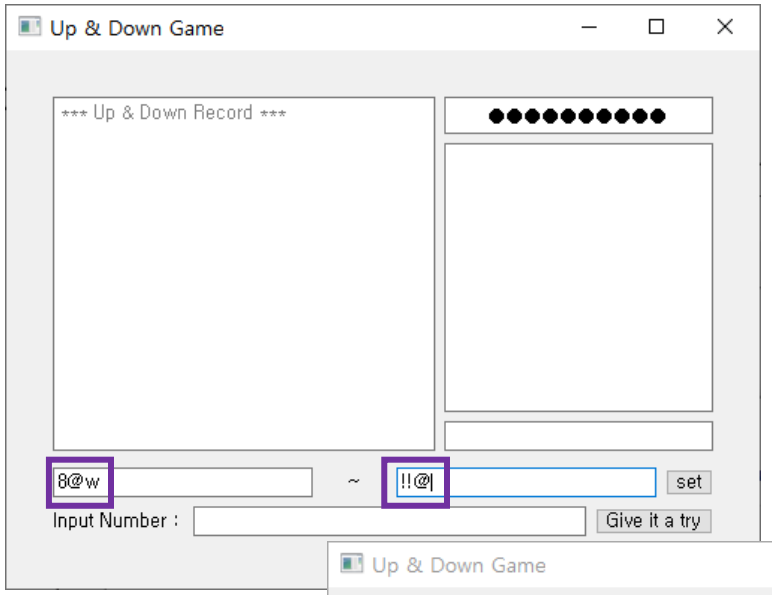
```
if(p.match(tryNum)!=None):
```

```
else:
    QMessageBox.about(self, 'NOTICE', 'Please input only NUMBER.')
```

QMessageBox를 이용하여
message를 출력

예외 처리

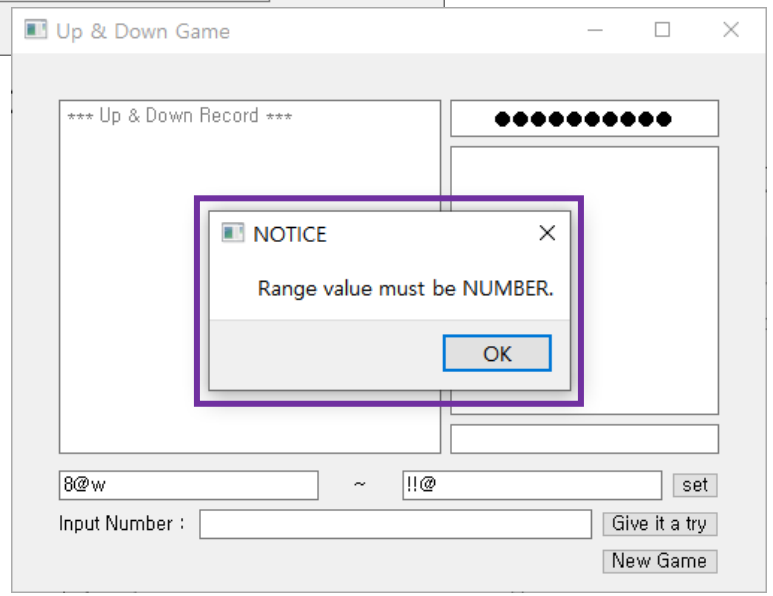
2. 입력된 값이 십진 정수가 아닐 때 – Range Number



game.py

```
def setClicked(self):
    p = re.compile(r'\b[0-9]+\b')
    if(p.match(self.startRangeInput.text())!=None and p.match(self.endRangeInput.text())!=None):
```

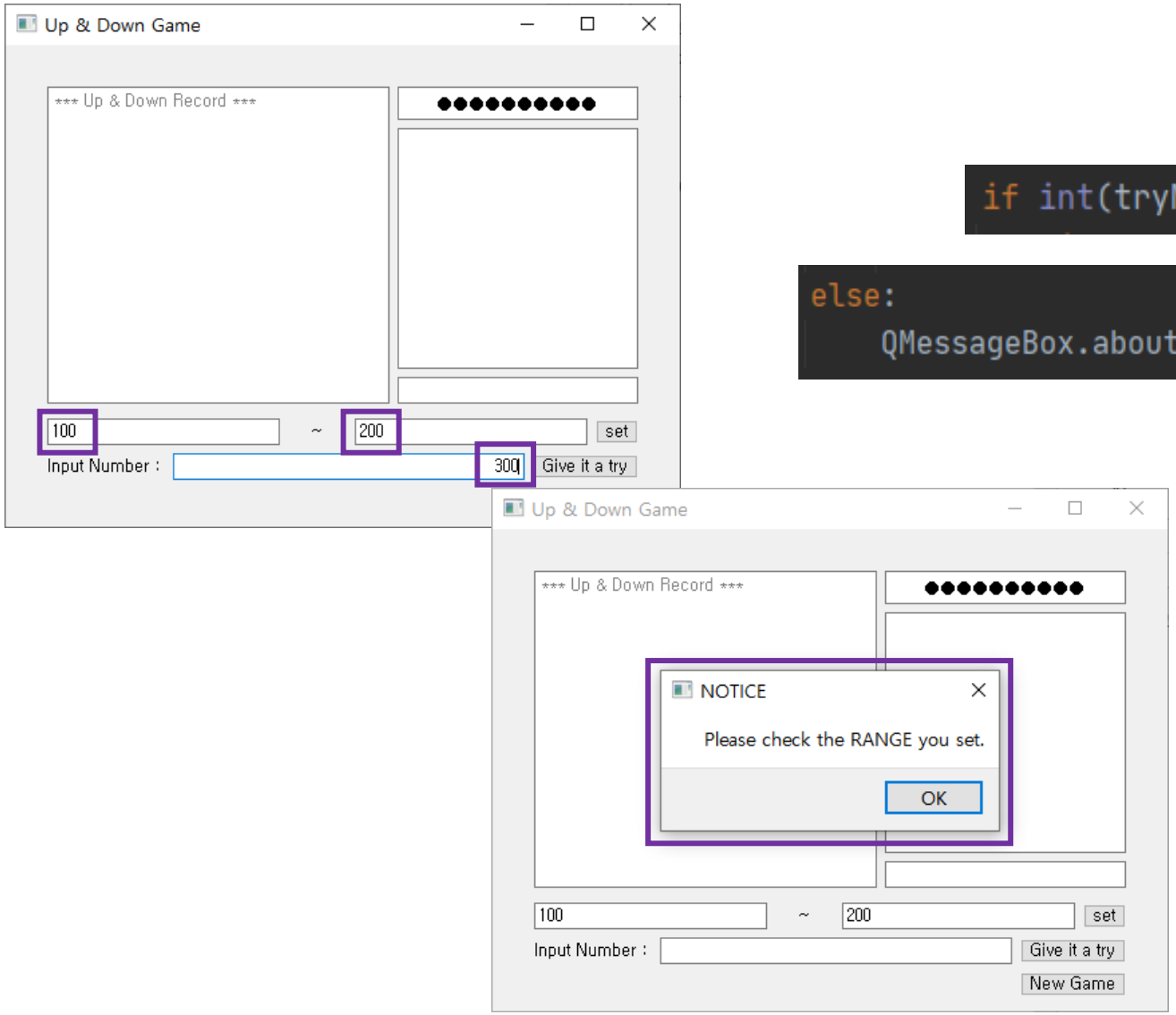
```
else:
    QMessageBox.about(self, 'NOTICE', 'Range value must be NUMBER.')
    self.startRangeInput.clear()
    self.endRangeInput.clear()
```



QMessageBox를 이용하여
message를 출력

예외 처리

3. 입력된 범위 밖의 숫자를 입력할 때



game.py

```
if int(tryNum) in range(self.startInput, self.endInput+1):
```

```
else:
    QMessageBox.about(self, 'NOTICE', 'Please check the RANGE you set.')
```

QMessageBox를 이용하여
message를 출력

감사합니다 :)