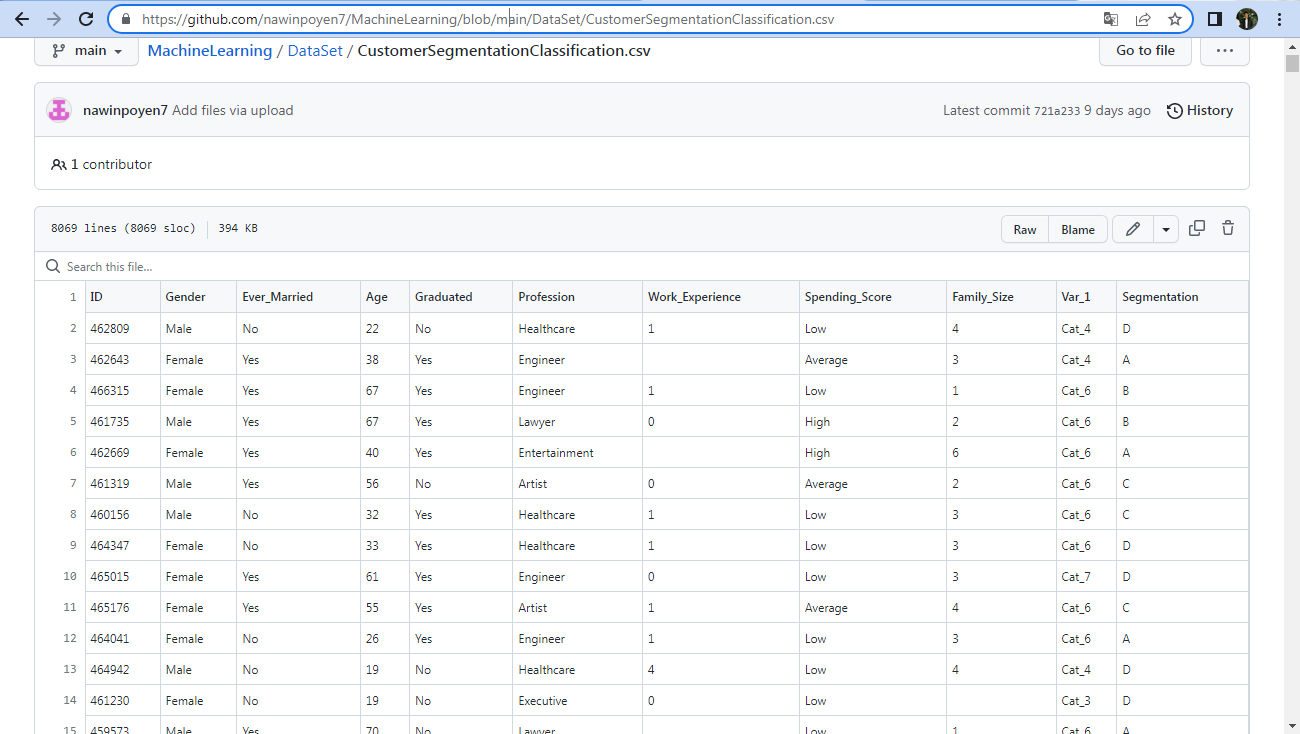
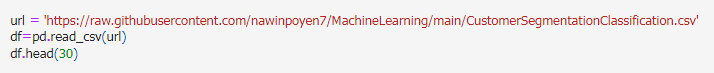
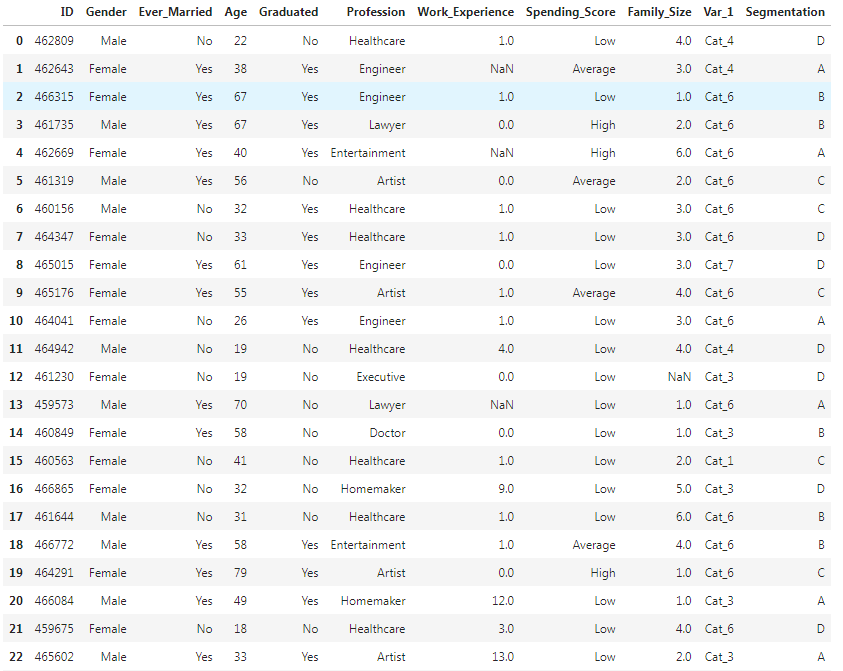
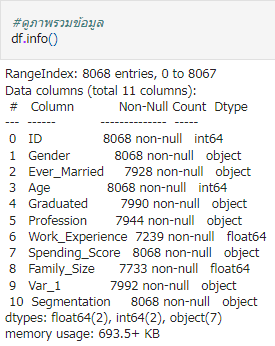


ให้ Data frame อ่านค่าจาก url ของไฟล์ CustomerSegmentationClassification.csvแล้วแสดงข้อมูล 30 ชุดแรก





จะเห็นได้ว่าข้อมูลของบาง Column นั้นไม่สมบูรณ์ เกิดการขาดหายไป ไม่สามารถนำมาใช้ได้ทันทีต้องทำการจัดการกับข้อมูลก่อน อาจจะจัดการโดยตัดชุดข้อมูลที่ไม่สมบูรณ์ครบถ้วนออกไป หรือ ทำการเติมข้อมูลลงในชุดข้อมูลที่ขาดหาย ( ตัวอย่างที่จะทำต่อจากนี้เป้นการเติมข้อมูล )



ดูลักษณะโดยรวมของข้อมูลจาก Data frame โดยที่

**RangeIndex: 8068 entries, 0 to 8067**

บอกว่าข้อมุลมีกี่ชุด ในที่นี้จำนวนข้อมูลทั้งหมดเท่ากับ 8068 ชุด

เริ่มจากชุดที่ 0 ถึง 8067

**Data columns (total 11 columns):**

บอกว่าข้อมุลมีกี่คอลัมน์ ในที่นี้มี 11 คอลัมน์

# Column Non-Null Count Dtype

-- --------- ------------ --------

**0 ID 8068 non-null int64**

คอลัมน์ที่ 0 ชื่อ ID ข้อมูล 8068 ชุด ไม่มีข้อมูลขาดหาย เป็นข้อมูลชนิด int64

**1 Gender 8068 non-null object**

คอลัมน์ที่ 1 ชื่อ Gender ข้อมูล 8068 ชุดไม่ขาดหาย เป็นข้อมูลชนิด object

**2 Ever\_Married 7928 non-null object**

คอลัมน์ที่ 2 ชื่อ Ever\_Marriedข้อมูล 7928 ชุดไม่ขาดหาย เป็นข้อมูลชนิด object

**3 Age 8068 non-null int64**

คอลัมน์ที่ 3 ชื่อ Age ข้อมูล 8068 ชุดไม่ขาดหาย เป็นข้อมูลชนิด int64

**4 Graduated 7990 non-null object**

คอลัมน์ที่ 4 ชื่อ Graduated ข้อมูล 7990 ชุดไม่ขาดหาย เป็นข้อมูลชนิด object

**5 Profession 7944 non-null object**

คอลัมน์ที่ 5 ชื่อ Profession ข้อมูล 7944 ชุดไม่ขาดหาย เป็นข้อมูลชนิด object

**6 Work\_Experience 7239 non-null float64**

คอลัมน์ที่ 6 ชื่อ Work\_Experience ข้อมูล 7239 ชุดไม่ขาดหาย เป็นข้อมูลชนิด float64

**7 Spending\_Score 8068 non-null object**

คอลัมน์ที่ 7 ชื่อ Spending\_Score ข้อมูล 8068 ชุดไม่ขาดหาย เป็นข้อมูลชนิด object

**8 Family\_Size 7733 non-null float64**

คอลัมน์ที่ 8 ชื่อ Family\_Size ข้อมูล 7733 ชุดไม่ขาดหาย เป็นข้อมูลชนิด float64

**9 Var\_1 7992 non-null object**

คอลัมน์ที่ 9 ชื่อ Var\_1 ข้อมูล 7992 ชุดไม่ขาดหาย เป็นข้อมูลชนิด object

**10 Segmentation 8068 non-null object**

คอลัมน์ที่ 10 ชื่อ Segmentation ข้อมูล 8068 ชุดไม่มีขาดหาย เป็นข้อมูลชนิด object

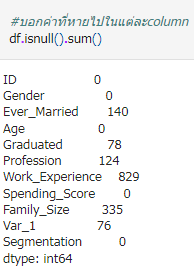
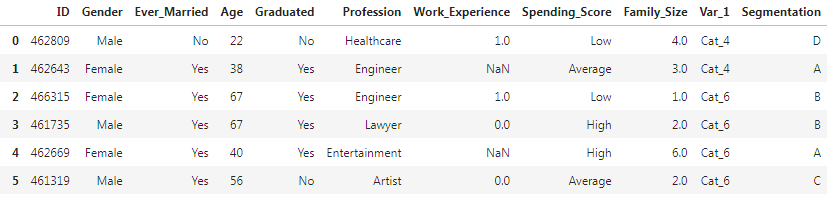
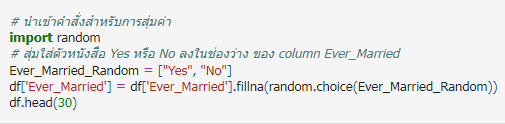
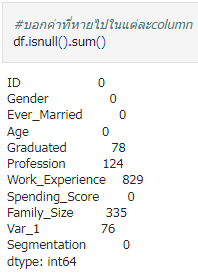
**dtypes: float64(2), int64(2), object(7)**

ประเภทข้อมูล: ข้อมูลชนิด float64 ( มีอยู่2คอลัมน์ ) ข้อมูลชนิด int64 ( มีอยู่2คอลัมน์ )

ข้อมูลชนิด object ( มีอยู่7คอลัมน์ )

**memory usage: 693.5+ KB**

หน่วยความจำที่ใช้: ขนาดใหญ่กว่า 693.5+ KB



จะเห็นได้ว่าที่ คอลัมน์ Ever\_Married ขึ้นเป็นเลข 0

แสดงว่าไม่มีข้อมูลที่ไม่สมบูรณ์เกิดเป็นช่องว่าง

ตรวจสอบหลังการใช้คำสั่ง ว่าในคอลัมน์ Ever\_Married ทำการเติมข้อมูลจนเต็มแล้วหรือยัง

สุ่มระหว่าง Yes หรือ No เติมลงใน คอลัมน์ Ever\_Married บริเวณข้อมูลที่ไม่สมบูรณ์เกิดเป็นช่องว่าง

ใช้คำสั่งให้ Data frame ทำการนับ ข้อมูลที่ไม่สมบูรณ์เกิดเป็นช่องว่าง

คอลัมน์ Ever\_Married มีทั้งหมด 140 ตำแหน่ง

คอลัมน์ Graduated มีทั้งหมด 78 ตำแหน่ง

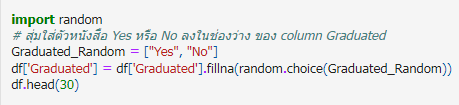
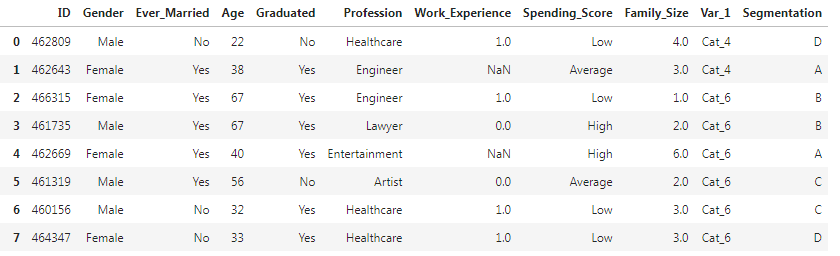
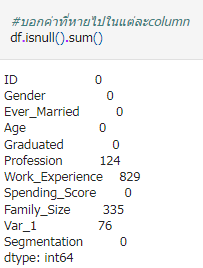
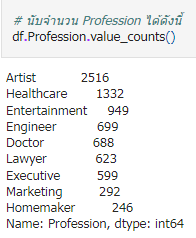
คอลัมน์ Profession มีทั้งหมด 124 ตำแหน่ง

คอลัมน์ Work\_Experience มีทั้งหมด 829 ตำแหน่ง

คอลัมน์ Family\_Size มีทั้งหมด 335 ตำแหน่ง

คอลัมน์ Var\_1 มีทั้งหมด 76 ตำแหน่ง

คอลัมน์ที่เหลือ ขึ้นเป็นเลข 0 แสดงว่าไม่มีข้อมูลที่ไม่สมบูรณ์เกิดเป็นช่องว่าง



เนื่องจากไม่รู้ว่าคอลัมน์ Professional มีค่าทั้งหมดกี่แบบ จึงใช้คำสั่งเพื่อนับว่า คอลัมน์ Professional มีค่าทั้งหมดกี่แบบ

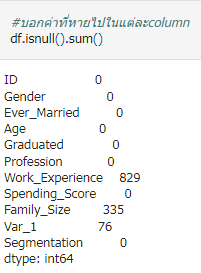
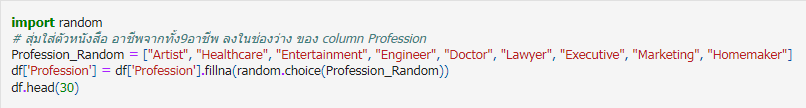
พบว่า คอลัมน์ Professional มีค่าทั้งหมด 9 แบบ (9 อาชีพ)

จะเห็นได้ว่าที่ คอลัมน์ Graduated ขึ้นเป็นเลข 0

แสดงว่าไม่มีข้อมูลที่ไม่สมบูรณ์เกิดเป็นช่องว่าง

ตรวจสอบหลังการใช้คำสั่ง ว่าในคอลัมน์ Graduated ทำการเติมข้อมูลจนเต็มแล้วหรือยัง

สุ่มระหว่าง Yes หรือ No เติมลงใน คอลัมน์ Graduated บริเวณข้อมูลที่ไม่สมบูรณ์เกิดเป็นช่องว่าง



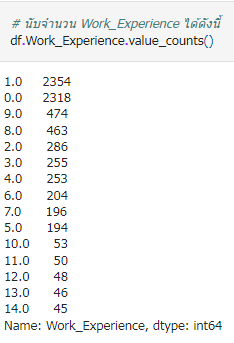
สุ่มระหว่าง Artist, Healthcare, Entertainment, Engineer, Doctor, Lawyer, Executive, Marketing และ Homemaker

เติมลงใน คอลัมน์ Professional บริเวณข้อมูลที่ไม่สมบูรณ์เกิดเป็นช่องว่าง

จะเห็นได้ว่าที่ คอลัมน์ Professional ขึ้นเป็นเลข 0

แสดงว่าไม่มีข้อมูลที่ไม่สมบูรณ์เกิดเป็นช่องว่าง

ตรวจสอบหลังการใช้คำสั่ง ว่าในคอลัมน์ Professional ทำการเติมข้อมูลจนเต็มแล้วหรือยัง

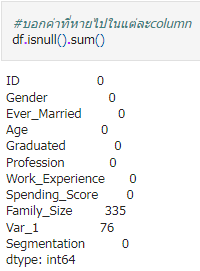
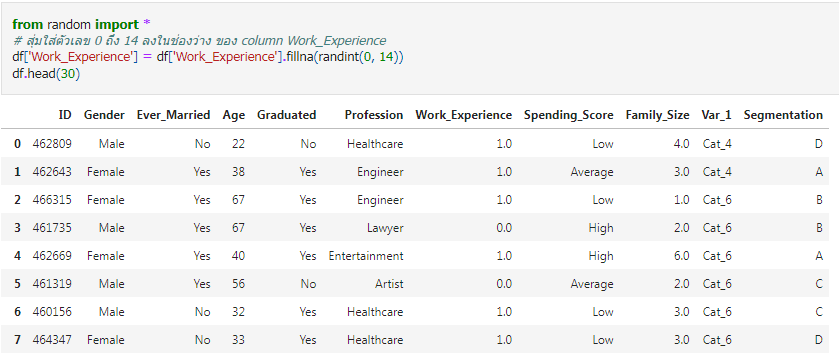


พบว่า คอลัมน์ Work\_Experience มีค่าทั้งหมด 15 แบบ (15 ค่า)

ค่าตั้งแต่ 0.0 ถึง 14.0

เนื่องจากไม่รู้ว่าคอลัมน์ Work\_Experience มีค่าทั้งหมดกี่แบบ จึงใช้คำสั่งเพื่อนับว่า คอลัมน์ Work\_Experience

มีค่าทั้งหมดกี่แบบ

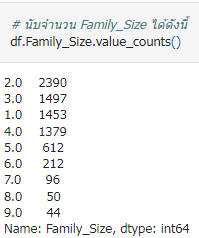
จะเห็นได้ว่าที่ คอลัมน์ Work\_Experience ขึ้นเป็นเลข 0

แสดงว่าไม่มีข้อมูลที่ไม่สมบูรณ์เกิดเป็นช่องว่าง

ตรวจสอบหลังการใช้คำสั่ง ว่าในคอลัมน์ Work\_Experience ทำการเติมข้อมูลจนเต็มแล้วหรือยัง

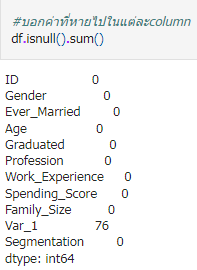
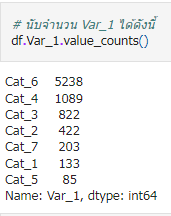
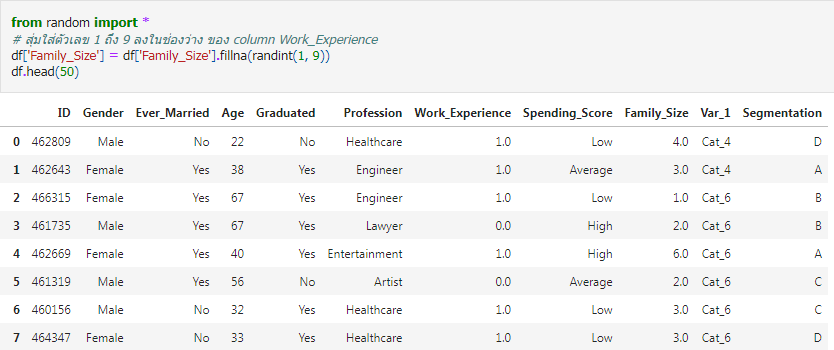
สุ่มเลขระหว่าง 0 ถึง 14 เติมลงใน คอลัมน์ Work\_Experience บริเวณข้อมูลที่ไม่สมบูรณ์เกิดเป็นช่องว่าง

เนื่องจากไม่รู้ว่าคอลัมน์ Family\_Size มีค่าทั้งหมดกี่แบบ จึงใช้คำสั่งเพื่อนับว่า คอลัมน์ Family\_Size มีค่าทั้งหมดกี่แบบ



พบว่า คอลัมน์ Family\_Size มีค่าทั้งหมด 9 แบบ (9 ค่า)

ค่าตั้งแต่ 1.0 ถึง 9.0

พบว่า คอลัมน์ Var\_1 มีค่าทั้งหมด 7 แบบ (7 ค่า)

ค่าตั้งแต่ Cat\_1 ถึง Cat\_7

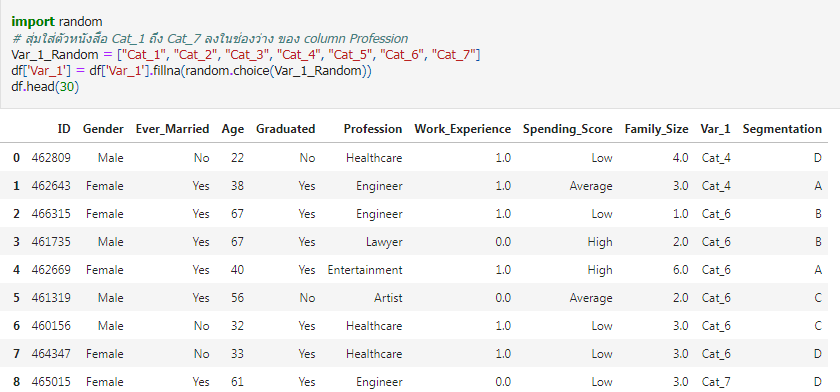
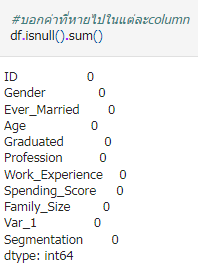
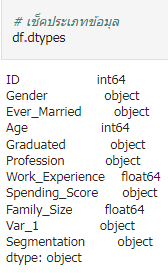
เนื่องจากไม่รู้ว่าคอลัมน์ Var\_1 มีค่าทั้งหมดกี่แบบ จึงใช้คำสั่งเพื่อนับว่า คอลัมน์ Var\_1 มีค่าทั้งหมดกี่แบบ

จะเห็นได้ว่าที่ คอลัมน์ Family\_Size ขึ้นเป็นเลข 0

แสดงว่าไม่มีข้อมูลที่ไม่สมบูรณ์เกิดเป็นช่องว่าง

ตรวจสอบหลังการใช้คำสั่ง ว่าในคอลัมน์ Family\_Size ทำการเติมข้อมูลจนเต็มแล้วหรือยัง

สุ่มเลขระหว่าง 1 ถึง 9 เติมลงใน คอลัมน์ Family\_Size บริเวณข้อมูลที่ไม่สมบูรณ์เกิดเป็นช่องว่าง



พบว่า ข้อมูลประเภท object นั้นเกิดจากค่าที่เป็นตัวอักษร หรือ ข้อความ

ข้อมูลประเภท float64 นั้นเกิดจากค่าตัวเลขที่เป็น ทศนิยม

ข้อมูล 2 ชนิดนี้ควรจะทำการแปลงชนิดข้อมูลก่อนเพื่อให้สามารถนำมาใช้งานได้สะดวก (ควรแปลงเป็น int64)

ตรวจสอบประเภทข้อมูลก่อนนำไปใช้งาน

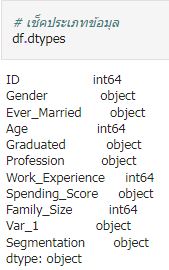
จะเห็นได้ว่าที่ คอลัมน์ Var\_1 และ คอลัมน์อื่นๆ ขึ้นเป็นเลข 0

แสดงว่าไม่มีข้อมูลที่ไม่สมบูรณ์เกิดเป็นช่องว่างในทุกๆคอลัมน์

ตรวจสอบหลังการใช้คำสั่ง ว่าในคอลัมน์ Var\_1 ทำการเติมข้อมูลจนเต็มแล้วหรือยัง

สุ่มระหว่าง Cat\_1, Cat\_2, Cat\_3, Cat\_4, Cat\_5, Cat\_6 และ Cat\_7 เติมลงใน คอลัมน์ Var\_1

บริเวณข้อมูลที่ไม่สมบูรณ์เกิดเป็นช่องว่าง



ทำการแปลง ข้อมูลของคอลัมน์ Spending\_Score ซึ่งมีข้อมูลชนิด object

จึงทำการแทนตัวหนังสือด้วยตัวเลข โดยให้ ‘High’ =2, ‘Average’=1และ ‘Low’=0

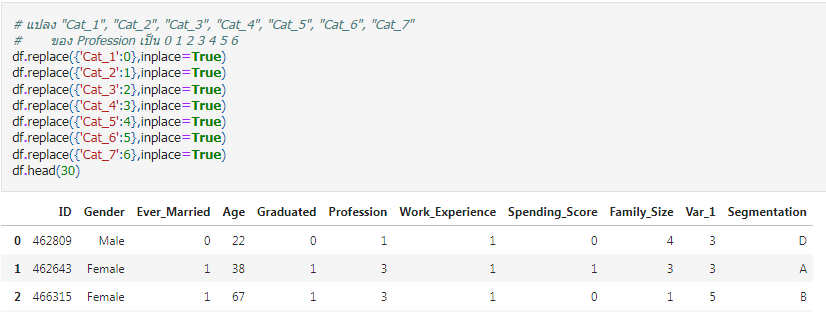
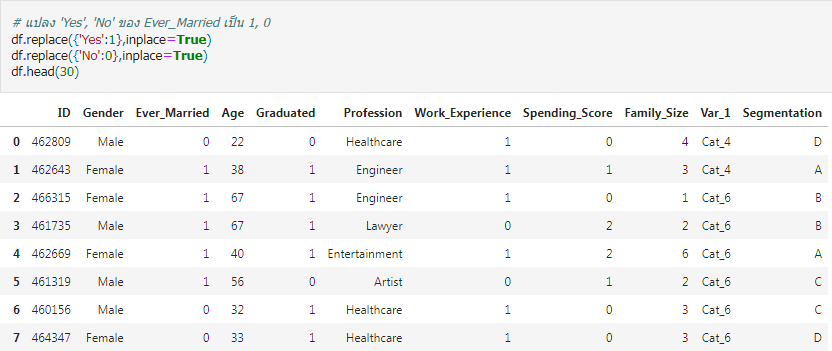
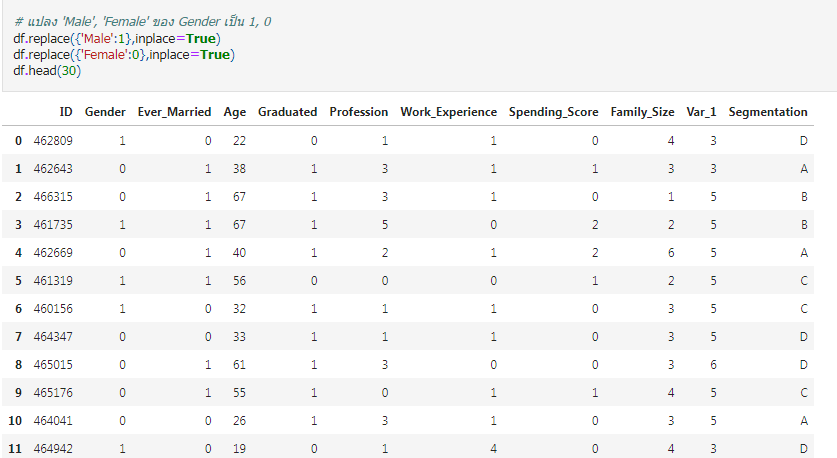
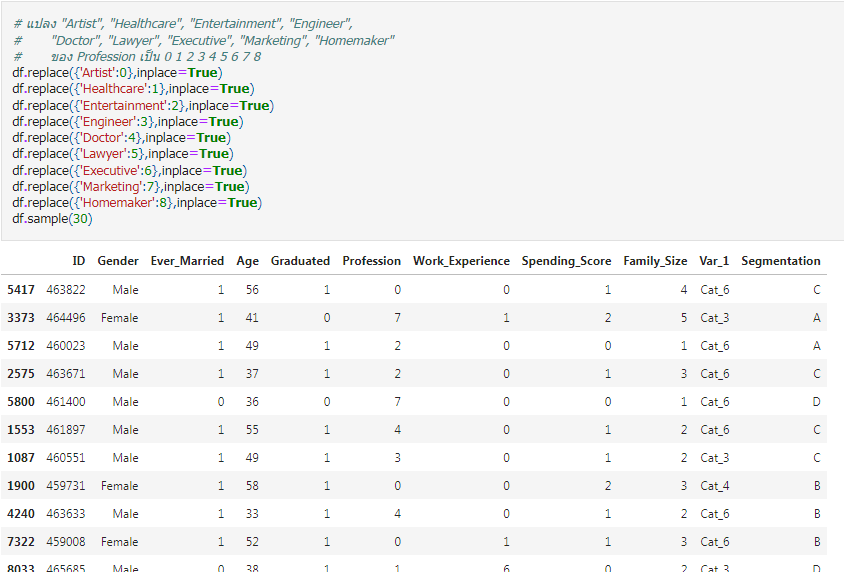
จะเห็นได้ว่าที่ คอลัมน์ Family\_Size และ Work\_Experience

ขึ้นเป็น int64 เป็นที่เรียบร้อยแล้ว

ตรวจสอบประเภทข้อมูลของ คอลัมน์ Family\_Size และ Work\_Experience ว่าแปลงเรียบร้อยหรือยัง

ทำการแปลง ข้อมูลของคอลัมน์ Family\_Size และ Work\_Experience ซึ่งมีข้อมูลชนิด float64

จึงทำการแปลงข้อมูลจากชนิด float64 เป็น int64



ทำการแปลง ข้อมูลของคอลัมน์ Profession ซึ่งมีข้อมูลชนิด object

จึงทำการแทนตัวหนังสือด้วยตัวเลข โดยให้ ‘Artist’=0, ‘Healthcare’=1, ‘Entertainment’=2, ‘Engineer’=3

‘Doctor’=4, ‘Lawyer’=5, ‘Executive’=6, ‘Marketing’=7 และ ‘Homemaker’=8

ทำการแปลง ข้อมูลของคอลัมน์ Gender ซึ่งมีข้อมูลชนิด object

จึงทำการแทนตัวหนังสือด้วยตัวเลข โดยให้ ‘Male’=1 และ ‘Female’=0

ทำการแปลง ข้อมูลของคอลัมน์ Ever\_Married และ Graduated ซึ่งมีข้อมูลชนิด object

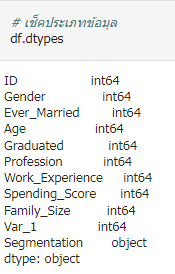
จึงทำการแทนตัวหนังสือด้วยตัวเลข โดยให้ ‘Yes’=1 และ ‘No’=0

ทำการแปลง ข้อมูลของคอลัมน์ Var\_1 ซึ่งมีข้อมูลชนิด object

จึงทำการแทนตัวหนังสือด้วยตัวเลข โดยให้ ‘Cat\_1’=0, ‘Cat\_2’=1, ‘Cat\_3’=2, ‘Cat\_4’=3

‘Cat\_5’=4, ‘Cat\_6’=5 และ ‘Cat\_7’=6

ตรวจสอบว่าคอลัมน์ที่จะใช้นั้นทำการแปลงชนิดข้อมูลเป็น int64 แล้วหรือยัง



พบว่าข้อมูลจากคอลัมน์ที่ต้องการใช้นั้นเป็นข้อมูลชนิด int64

พร้อมใช้งานได้อย่างสะดวกสบาย

(ในที่นี้ ไม่ใช้ ข้อมูลจากคอลัมน์ ID ส่วนข้อมูลจากคอลัมน์Segmentationนั้น ถึงจะไม่ได้แปลงแต่ยังต้องใช้เพื่อการจำแนกข้อมูล ในการทำต่อไป)

บันทึกข้อมูลทั้งหมดชื่อว่า DataChange.csv (ไฟล์สกุล csv)

