

Map(function,iterables)

In [1]:

1	p=[i for i in range(1000000	(http://www.mobilenumbertracker.com/MobileNum
2	s=lambda x:x**2	
3		
4	list(map(s,p))	

Out[1]:

[0,

1,

4,

9,

16,

25,

36,

49,

64,

81,

100,

121,

144,

169,

196,

225,

256,

289,

324,

361,

400,

441,

484,

529,

576,

625,

676,

729,

784,

841,

900,

961,

1024,

1089,

1156,

1225,

1296,

1369,

1444,

1521,

1600,

1681,

1764,

1849,

1936,

2025,

2116,

2209,

2304,

2401,

2500,

2601,

2704,

2809,

2916,

3025,

3136,

3249,

3364,

3481,

3600,

3721,

3844,

3969,

4096,

4225,

4356,

4489,

4624,

4761,

4900,

5041,

5184,

5329,

5476,

5625,

5776,

5929,

6084,

6241,

6400,

6561,

6724,

6889,

7056,

7225,

7396,

7569,

7744,

7921,

8100,

8281,

8464,

8649,

8836,

9025,

9216,

9409,

9604,

9801,

10000,

10201,

10404,

10609,

10816,

11025,

11236,

11449,

11664,

11881,

12100,

12321,

12544,

12769,

12996,

13225,

13456,

13689,

13924,

14161,

14400,

14641,

14884,

15129,

15376,

15625,

15876,

16129,

16384,

16641,

16900,

17161,

17424,

17689,

17956,

18225,

18496,

18769,

19044,

19321,

19600,

19881,

20164,

20449,

20736,

21025,

21316,

21609,

21904,

22201,

22500,

22801,

23104,

23409,

23716,

24025,

24336,

24649,

24964,

25281,

25600,

25921,

26244,

26569,

26896,

27225,

27556,

27889,

28224,

28561,

28900,

29241,

29584,

29929,

30276,

30625,

30976,

31329,

31684,

32041,

32400,

32761,

33124,

33489,

33856,

34225,

34596,

34969,

35344,

35721,

36100,

36481,

36864,

37249,

37636,

38025,

38416,

38809,

39204,

39601,

40000,

40401,

40804,

41209,

41616,

42025,

42436,

42849,

43264,

43681,

44100,

44521,

44944,

45369,

45796,

46225,

46656,

47089,

47524,

47961,

48400,

48841,

49284,

49729,

50176,

50625,

51076,

51529,

51984,

52441,

52900,

53361,

53824,

54289,

54756,

55225,

55696,

56169,

56644,

57121,

57600,

58081,

58564,

59049,

59536,

60025,

60516,

61009,

61504,

62001,

62500,

63001,

63504,

64009,

64516,

65025,

65536,

66049,

66564,

67081,

67600,

68121,

68644,

69169,

69696,

70225,

70756,

71289,

71824,

72361,

72900,

73441,

73984,

74529,

75076,

75625,

76176,

76729,

77284,

77841,

78400,

78961,

79524,

80089,

80656,

81225,

81796,

82369,

82944,

83521,

84100,

84681,

85264,

85849,

86436,

87025,

87616,

88209,

88804,

89401,

90000,

90601,

91204,

91809,

92416,

93025,

93636,

94249,

94864,

95481,

96100,

96721,

97344,

97969,

98596,

99225,

100000,

100801,

101604,

102409,

103216,

104025,

104836,

105649,

106464,

107281,

108100,

108921,

109744,

110569,

111396,

112225,

113056,

113889,

114724,

115561,

116400,

117241,

118084,

118929,

119776,

120625,

121476,

122329,

123184,

124041,

124900,

125761,

126624,

127489,

128356,

129225,

130096,

130969,

131844,

132721,

133600,

134481,

135364,

136249,

137136,

138025,

138916,

139809,

140704,

141601,

142500,

143401,

144304,

145209,

146116,

147025,

147936,

148849,

149764,

150681,

151600,

152521,

153444,

154369,

155296,

156225,

157156,

158089,

159024,

159961,

160900,

161841,

162784,

163729,

164676,

165625,

166576,

167529,

168484,

169441,

170400,

171361,

172324,

173289,

174256,

175225,

176196,

177169,

178144,

179121,

180100,

181081,

182064,

183049,

184036,

185025,

186016,

187009,

188004,

189001,

190000,

191001,

192004,

193009,

194016,

195025,

196036,

197049,

198064,

199081,

200100,

201121,

202144,

203169,

204196,

205225,

206256,

207289,

208324,

209361,

210400,

211441,

212484,

213529,

214576,

215625,

216676,

217729,

218784,

219841,

220900,

221961,

223024,

224089,

225156,

226225,

227296,

228369,

229444,

230521,

231600,

232681,

233764,

234849,

235936,

237025,

238116,

239209,

240304,

241401,

242500,

243601,

244704,

245809,

246916,

248025,

249136,

250249,

251364,

252481,

253600,

254721,

255844,

256969,

258096,

259225,

260356,

261489,

262624,

263761,

264900,

266041,

267184,

268329,

269476,

270625,

271776,

272929,

274084,

275241,

276400,

277561,

278724,

279889,

281056,

282225,

283396,

284569,

285744,

286921,

288100,

289281,

290464,

291649,

292836,

294025,

295216,

296409,

297604,

298801,

300000,

301201,

302404,

303609,

304816,

306025,

307236,

308449,

309664,

310881,

312100,

313321,

314544,

315769,

316996,

318225,

319456,

320689,

321924,

323161,

324400,

325641,

326884,

328129,

329376,

330625,

331876,

333129,

334384,

335641,

336900,

338161,

339424,

340689,

341956,

343225,

344496,

345769,

347044,

348321,

349600,

350881,

352164,

353449,

354736,

356025,

357316,

358609,

359904,

361201,

362500,

363801,

365104,

366409,

367716,

369025,

370336,

371649,

372964,

374281,

375600,

376921,

378244,

379569,

380896,

382225,

383556,

384889,

386224,

387561,

388900,

390241,

391584,

392929,

394276,

395625,

396976,

398329,

399684,

401041,

402400,

403761,

405124,

406489,

407856,

409225,

410596,

411969,

413344,

414721,

416100,

417481,

418864,

420249,

421636,

423025,

424416,

425809,

427204,

428601,

430000,

431401,

432804,

434209,

435616,

437025,

438436,

439849,

441264,

442681,

444100,

445521,

446944,

448369,

449796,

451225,

452656,

454089,

455524,

456961,

458400,

459841,

461284,

462729,

464176,

465625,

467076,

468529,

469984,

471441,

472900,

474361,

475824,

477289,

478756,

480225,

481696,

483169,

484644,

486121,

487600,

489081,

490564,

492049,

493536,

495025,

496516,

498009,

499504,

501001,

502500,

504001,

505504,

507009,

508516,

510025,

511536,

513049,

514564,

516081,

517600,

519121,

520644,

522169,

523696,

525225,

526756,

528289,

529824,

531361,

532900,

534441,

535984,

537529,

539076,

540625,

542176,

543729,

545284,

546841,

548400,

549961,

551524,

553089,

554656,

556225,

557796,

559369,

560944,

562521,

564100,

565681,

567264,

568849,

570436,

572025,

573616,

575209,

576804,

578401,

580000,

581601,

583204,

584809,

586416,

588025,

589636,

591249,

592864,

594481,

596100,

597721,

599344,

600969,

602596,

604225,

605856,

607489,

609124,

610761,

612400,

614041,

615684,

617329,

618976,

620625,

622276,

623929,

625584,

627241,

628900,

630561,

632224,

633889,

635556,

637225,

638896,

640569,

642244,

643921,

645600,

647281,

648964,

650649,

652336,

654025,

655716,

657409,

659104,

660801,

662500,

664201,

665904,

667609,

669316,

671025,

672736,

674449,

676164,

677881,

679600,

681321,

683044,

684769,

686496,

688225,

689956,

691689,

693424,

695161,

696900,

698641,

700384,

702129,

703876,

705625,

707376,

709129,

710884,

712641,

714400,

716161,

717924,

719689,

721456,

723225,

724996,

726769,

728544,

730321,

732100,

733881,

735664,

737449,

739236,

741025,

742816,

744609,

746404,

748201,

750000,

751801,

753604,

755409,

757216,

759025,

760836,

762649,

764464,

766281,

768100,

769921,

771744,

773569,

775396,

777225,

779056,

780889,

782724,

784561,

786400,

788241,

790084,

791929,

793776,

795625,

797476,

799329,

801184,

803041,

804900,

806761,

808624,

810489,

812356,

814225,

816096,

817969,

819844,

821721,

823600,

825481,

827364,

829249,

831136,

833025,

834916,

836809,

838704,

840601,

842500,

844401,

846304,

848209,

850116,

852025,

853936,

855849,

857764,

859681,

861600,

863521,

865444,

867369,

869296,

871225,

873156,

875089,

877024,

878961,

880900,

882841,

884784,

886729,

888676,

890625,

892576,

894529,

896484,

898441,

900400,

902361,

904324,

906289,

908256,

910225,

912196,

914169,

916144,

918121,

920100,

922081,

924064,

926049,

928036,

930025,

932016,

934009,

935996,

937989,

939984,

941981,

943984,

945989,

947996,

949961,

951924,

953889,

955856,

957825,

959796,

961769,

963744,

965721,

967700,

969681,

971664,

973649,

975636,

977625,

979616,

981609,

983604,

985601,

987600,

989601,

991604,

993609,

995616,

997625,

1000000

]

Filter(function,iterables)

In [2]:

1	p
---	---

Out[2]:

[0,

1,

2,

3,

4,

5,

6,

7,

8,

9,

10,

11,

12,

13,

14,

15,

16,

17,

18,

19,

20,

21,

22,

23,

24,

25,

26,

27,

28,

29,

30,

31,

32,

33,

34,

35,

36,

37,

38,

39,

40,

41,

42,

43,

44,

45,

46,

47,

48,

49,

50,

51,

52,

53,

54,

55,

56,

57,

58,

59,

60,

61,

62,

63,

64,

65,

66,

67,

68,

69,

70,

71,

72,

73,

74,

75,

76,

77,

78,

79,

80,

81,

82,

83,

84,

85,

86,

87,

88,

89,

90,

91,

92,

93,

94,

95,

96,

97,

98,

99,

100,

101,

102,

103,

104,

105,

106,

107,

108,

109,

110,

111,

112,

113,

114,

115,

116,

117,

118,

119,

120,

121,

122,

123,

124,

125,

126,

127,

128,

129,

130,

131,

132,

133,

134,

135,

136,

137,

138,

139,

140,

141,

142,

143,

144,

145,

146,

147,

148,

149,

150,

151,

152,

153,

154,

155,

156,

157,

158,

159,

160,

161,

162,

163,

164,

165,

166,

167,

168,

169,

170,

171,

172,

173,

174,

175,

176,

177,

178,

179,

180,

181,

182,

183,

184,

185,

186,

187,

188,

189,

190,

191,

192,

193,

194,

195,

196,

197,

198,

199,

200,

201,

202,

203,

204,

205,

206,

207,

208,

209,

210,

211,

212,

213,

214,

215,

216,

217,

218,

219,

220,

221,

222,

223,

224,

225,

226,

227,

228,

229,

230,

231,

232,

233,

234,

235,

236,

237,

238,

239,

240,

241,

242,

243,

244,

245,

246,

247,

248,

249,

250,

251,

252,

253,

254,

255,

256,

257,

258,

259,

260,

261,

262,

263,

264,

265,

266,

267,

2

```
In [3]: 1 d=lambda t: t if t%2==0 else None
        2 list(filter(d,p))
```

```
Out[3]: [2,
         4,
         6,
         8,
         10,
         12,
         14,
         16,
         18,
         20,
         22,
         24,
         26,
         28,
         30,
         32,
         34,
         36,
         38,
         ..
```

```
In [4]: 1 def find_vowels_consonants(char):
        2     predefineList=['a','e','i','o','u']
        3     if char in predefineList:
        4         return False #'vowel'
        5     else:
        6         return True #'consonants'
        7
        8 to=list('Piyush')
        9 result=filter(find_vowels_consonants,to)
       10 list(result)
```

```
Out[4]: ['P', 'y', 's', 'h']
```

```
In [ ]: 1
```

```
In [ ]: 1
```

```
In [ ]: 1
```

Reduce(function,iterables)

```
In [5]: 1 o= lambda a,b: a+b
```

```
In [6]: 1 o(2,4)
```

```
Out[6]: 6
```

```
In [7]: 1 from functools import reduce
```

In [8]:

1	reduce(o,p)
---	-------------

Out[8]: 499999500000 (http://www.mobilenumbertracker.com/MobileNumberLocation?number=499999500000&utm_source=http://localhost:8889/notebooks/Map%20reduce%20%26%20filter.ipynb&utm_medium=referral&utm_campaign=chrome-plugin)

In []:

1	
---	--

In []:

1	
---	--