session_3 > employee_data.html

```
243
244  # Tasks:
245  # 1. Print the name of the person who has the highest salary at the company.
246
247  employee_with_highest_salary = employees[0]
248  for employee in employees:
249      if(employee["salary"] > employee_with_highest_salary["salary"]):
250          employee_with_highest_salary = employee
251
252  print("\nTask 1:")
253  print(employee_with_highest_salary["first_name"],employee_with_highest_salary["last_name"])
254
255  # 2. Print the combined years of experience of all employees at the company.
256
257  combined_years_of_experience = 0
258  for employee in employees:
259      combined_years_of_experience += employee["years_of_experience"]
260
261  print("\nTask 2:")
262  print("Combined years of experience of all employees at the company:", combined_years_of_experience, "years")
263
264  # 3. Some people don't have an email address - collect their details into a new list!
265
266  employee_with_no_emails_address = []
267  for employee in employees:
268      if(not employee["email"]):
269          employee_with_no_emails_address.append(employee)
270
271  print("\nTask 3:")
```

session_3 > employee_data.html

```
270
271  print("\nTask 3:")
272  print(employee_with_no_emails_address)
273
274  # 4. Which one costs more for the company - Product department salaries or Business department salaries?
275
276  product_salaries = sum(employee["salary"] for employee in employees if employee["department"] == "Product")
277  business_salaries = sum(employee["salary"] for employee in employees if employee["department"] == "Business")
278  print("\nTask 4:")
279  print("The total Product department salaries:", product_salaries)
280  print("The total Business department salaries:",business_salaries)
281
282  if product_salaries > business_salaries:
283      print("Product department salaries cost more for the company.")
284  else:
285      print("Business department salaries cost more for the company.")
286
287  # Extensions: 5. What is the average salary for people over 30 years of age?
288
289  people_over_30 = [employee for employee in employees if employee["age"] > 30]
290
291  total_salary_over_30 = sum(employee["salary"] for employee in people_over_30)
292
293  if len(people_over_30) > 0:
294      average_salary_over_30 = total_salary_over_30 / len(people_over_30)
295      print("\nExtension 5:")
296      print(f"Average salary for people over 30 years of age: ${average_salary_over_30:.2f}")
297  else:
298      print("No employees over 30 years of age.")
```

session_3 > employee_data.html

```
292
293 if len(people_over_30) > 0:
294     average_salary_over_30 = total_salary_over_30 / len(people_over_30)
295     print("\nExtension 5:")
296     print(f"Average salary for people over 30 years of age: ${average_salary_over_30:.2f}")
297 else:
298     print("No employees over 30 years of age.")
299
300 # Extension 6. Create a new dict and calculate how many people are working with certain job titles.
301 # (HARD) Example:
302 #{"Project Manage": 4, "Machine Learning Engineer": 3, ...}
303
304 job_title_count = {}
305
306 for employee in employees:
307     job_title = employee["job_title"]
308     if job_title in job_title_count:
309         job_title_count[job_title] += 1
310     else:
311         job_title_count[job_title] = 1
312
313 print("\nExtension 6:")
314 print("Count of people by job title:")
315 for job_title, count in job_title_count.items():
316     print(f"{job_title}: {count}")
317
```