

Bu metin, "Yaşam Boyu Anaokulu / Lifelong Kindergarten"
kitabından alıntıdır.

BÖLÜM 2: Projeler

Yaşam Boyu Anaokulu

Lifelong Kindergarten

Projeler, Tutku, Akranlar ve Oyun
Yoluyla Yaratıcılığı Geliştirme

Mitchel Resnick

© 2017

Çevirmen: Gamze Sart,
Burcu Çetin, Cem Aşkın

abayayin.com/yasamboyuanaokulu/



ÜRETİCİLER

Ocak 2009'da, MIT kampüsündeki büyük bir konferans salonunda, Amerika Birleşik Devletleri'nin 44. başkanı olarak Barack Obama'nın tanıtılışını izledim. Salon 500'den fazla kişiyle doluydu ve salonun önündeki iki büyük ekranda Obama'nın tanıtım konuşmasının bir videosu gösteriliyordu. Salonun MIT bilim insanları ve mühendisleri ile dolu olduğu göz önüne alındığında, en güçlü tepkinin Obama'nın şunu ilan ettiğinde geldiğini duymak şaşırtıcı olmayacaktır: "Bilimi hak ettiği yere geri getireceğiz." Salon alkıştan inledi.

Ancak tanıtım konusundaki dikkatimi çeken nokta bu değildi. Benim için en çok hafızamda kalacak olan Obama'nın, "Bizleri refah ve özgürlüğe giden uzun ve inişli çıkışlı yolda sona ulaştıranlar risk alanlar, uygulayıcılar ve üreticilerdir: Kadın olsun, erkek olsun, bazıları kutsandır ama sıklıkla emekleri görülmez," dediği andı.

Risk alanlar. Uygulayıcılar. Üreticiler. Bunlar, yaratıcı düşünürler olan X grubu öğrenciler. Tarih boyunca ekonomik, teknolojik, siyasi ve kültürel değişimin arkasındaki itici güç oldular. Günümüzde herkes, risk alan, uygulayan, üreten bireyler olmalı ve bunu tarihin gidişatını değiştirmek için değil, kendi yaşamlarına yön vermek için yapmalıdır.

Obama, kullandığı üreticiler kavramı ile kültürümüz üzerinden henüz yayılmaya başlamış bir harekete örtülü bir gönderme yapıyordu: Maker Hareketi. Bu hareket, bir şeyler üretmekten tutku duyan kişiler arasında bodrumlarda, garajlarda ve toplum merkezlerinde fikirlerini ve yaratımlarını birbirleriyle paylaşan kişiler arasında tabandan bir hareket olarak başladı. 2005'te harekete ivme kazandıran, Dale Dougherty'nin piyasaya sürdüğü Make: dergisiydi. Bu dergi, bir şeyler inşa etmenin, yaratmanın ve icat etmenin mutluluğunu övüyordu. Dergi, üretmeyi demokratikleştirmeyi ve herkesin kendi başına yapabileceği etkinliklere nasıl dahil olacağını göstermeyi amaçlıyordu. Birinci sayı, havadan

fotoğraflar çekmek için kullanılacak bir uçurtma yapmaya, birayı soğutmak için bir termoelektrik fiçı örtü yapmaya ve geceleri dinamik ışık desenleri oluşturmak için parıldayan çubuklar hazırlamaya yönelik talimatları içererek, “sıradan insanların garajlarında ve bahçelerinde yapacakları şaşırtıcı şeyleri” anlatıyordu.

Ertesi yıl, 2006’da Dale, “buluş, yaratıcılık ve beceriklilik için aile dostu bir festival” olarak tanımladığı ilk Maker Fuarı’nı organize etti. Takı tasarımı, mobilya tasarımı, robot yapma gibi hayal edebileceğiniz neredeyse her şeyi üreteceğiniz sergiler ve atölyeler vardı. Son on yılda dünyanın çeşitli noktalarında milyonlarca mühendis, sanatçı, tasarımcı, girişimci, eğitimci, ebeveyn ve çocuğun ilgisini çeken yüzlerce Maker fuarı yapıldı.

Birçok kişi için Maker Hareketi’nin esas çekiciliği teknoloji alanındaydı. İnsanların fiziksel nesnelere tasarlayıp üretmesine ve özelleştirmesine olanak sağlayan 3 boyutlu yazıcılar ve lazer kesiciler gibi yeni teknolojiler yaygınlaşıyordu. Birçok kişi, bu teknolojilerin kendisine sağlayacağı iş potansiyeli konusunda heyecanlanıyor ve Maker Hareketi’nin küçük işletmelerin (hatta bireylerin) daha önce ölçekli ekonomiler ve büyük fabrikalar gerektiren ürünler üretebilecekleri yeni bir sanayi devrimi başlatacağına inanıyordu.

Ben Maker Hareketi’ne farklı sebeplerden dolayı ilgi duyuyorum. İnanıyorum ki hareket sadece teknolojik ve ekonomik bir hareket olmakla kalmayıp insanların yaratıcı öğrenme deneyimlerine yeni yollar açacak bir öğrenme hareketi yaratma potansiyeline de sahip olacaktır. İnsanlar kendileri üretip yaratırken, yaratıcı düşüncüler olarak gelişme fırsatlarına da sahip olacak. Sonuçta, yaratıcılık işinin özünde yaratmak vardır.

Belki de en önemlisi, Maker Hareketi’nin insanları, yaratıcı öğrenmenin dört P’sinden ilki olan “Projeler”de çalışmaya teşvik etmesidir. Make: dergisindeki makaleler ve Maker fuarındaki sergiler sadece üretim tekniklerini öğretmez; aynı zamanda insanla-

ra anlamlı projeler üzerinde çalışırken yeni fikirler, beceriler ve stratejiler öğrenmeleri için gereken proje tabanlı bir öğrenme yaklaşımını da sunar. Dale Dougherty, projelerden “üretim sürecinin temel birimleri” olarak bahseder.

Ben büyürken projelerin gücünü kişisel olarak deneyimledim. Çocukken, sporun her çeşidinden hoşlanırdım: Beysbol, basketbol, tenis ve daha fazlası. Ancak sportif oyunlar oynamanın ötesinde, “spor yapma” fikri hoşuma gidiyordu. Kardeşim ve kuzenimle oynamak için sürekli yeni sporlar icat ediyordum. İnşa etmek ve oyun oynamak için bir arka bahçemiz olduğundan ve arka bahçemizi projelerim için bir çalışma alanına dönüştürmeme izin veren ebeveynlerim olduğu için şanslıydım.

Bir yaz günü, kendi minyatür golf sahamı oluşturmak için arka bahçeyi kazdım. Benim için sürekli bir öğrenme deneyimi oldu. Golf delikleri için başlangıçta zemin üzerinde basit delikler kazarak işe başladım, ancak zamanla deliklerin şekillerini kaybettiğini gördüğümde, deliklere alüminyum teneke kutuları yerleştirmeye başladım. Yağmur yağana kadar bu işe yaradı ancak teneke kutuları dolduran yağmur suyundan kurtulmak zordu. Çözümü teneke kutuların her iki ucunu toprağa gömmeden önce kesmekte buldum, böylece su alttan dışarı akabilecekti.

Mini golf sahama duvarlar ve engeller ekledikçe, topun oralarından nasıl sekeceğini hesaplamam gerekti. Bu bana çarpışma fiziğini öğrenmem için motive edici bir bağlam sağladı.

Açıları hesaplamak ve ölçmek için saatler harcadım. Öyle ki bir golf topunu engellerin üzerinden atıyor ve deliğe sıçratıyordum. Bu deneyim sınıfta gördüğüm fen derslerinden kesinlikle daha akılda kalıcıydı.

Süreç boyunca sadece minyatür bir golf sahamı oluşturma sürecini değil, herhangi bir şeyi üretmek için genel bir sürecin nasıl olması gerektiği hakkında bir fikir geliştirmeye başladım: Öncül

fikirle nasıl başlanır, hazırlık planları nasıl geliştirilir, ilk versiyon nasıl oluşturulur, denenir, başkalarından denemeleri istenir, ne yapıp yapmadığına bakılarak planlar gözden geçirilir ve bu işlem tekrar tekrar nasıl devam eder? Projem üzerinde çalışarak Yaratıcı Öğrenme Spirali konusunda deneyim kazandım.

Bu tür projeler sayesinde kendimi bir şeyler üreten ve yaratan birisi olarak görmeye başladım. Dünyadaki şeylere yeni bir gözle, nasıl yapıldıklarını merak ederek bakmaya başladım. Bir golf topu nasıl yapılmıştı ya da bir golf kulübü? Daha başka neler üretebileceğimi sorgulamaya başladım.

Eğer Make: İnternet sitesinde bugün arama yaparsanız (makezine.com), minyatür golf projelerini tarif eden çok sayıda makale bulabilirsiniz, “Kendi Masaüstü Mini Golfünüzü Tasarlayın” ve “Urban Putt: Miniature Golf 2.0.” gibi. Kendi minyatür golf sahamı yaklaşık 50 yıl önce inşa ettiğimden beri teknoloji çok gelişti. Artık 3 boyutlu bir yazıcı veya lazer kesici ile özel olarak tasarlanmış engeller üretmek mümkün ve artık engellerin içine golf topu bir engelle karşılaştığında motorları harekete geçirecek veya LED’leri yakacak sensörler yerleştirmek mümkün.

Hâlâ küçük bir çocuk olarak inşa ettiğim “demode” minyatür golf sahama gurur duyarım. Ancak aynı zamanda, yeni teknolojilerin çocukların yaratabileceği projelerin türlerini genişletebileceği ve daha fazla çocuğa bir şeyler üretebileceği konusunda ilham vermesi beni heyecanlandırıyor.

ÜRETEREK ÖĞRENME

Yıllar boyunca birçok eğitimci ve araştırmacı yaparak öğrenmenin, insanların yapma eylemine aktif olarak dahil edilmesinin daha iyi olduğunu ve insanların uygulamalı eğitimler sayesinde daha iyi öğreneceklerini savundu.

Fakat Maker Hareket kültüründe bir şeyi sadece yapmak yeterli değil, aynı zamanda bir şey üretmek zorundasınız. Üretici etiğine göre, en kıymetli öğrenme deneyimlerini tasarım, inşa ve yaratma sürecine aktif olarak dahil edildiğinizde kazanırsınız, yani üreterek öğrendiğinizde.

Üretme ve öğrenme arasındaki bağlantıları daha iyi kavramak ve üretmek öğrenmeyi nasıl destekleyeceğinizi görmek için Seymour Papert'in çalışmalarından daha iyi bir kaynak bulamazsınız. Seymour'la MIT'de uzun yıllar çalışma fırsatı bulduğum için şanslıydım. Seymour, üretmek öğrenme konusunda herkesten fazla entellektüel temel geliştirdi, teknolojileri ve stratejileri de bunu desteklemek için kanalize etti. Nitekim Seymour, Maker Hareketi'nin koruyucu azizi olarak düşünölmelidir.

Seymour öğrenmeyi tüm boyutlarıyla severdi: Anlamak, desteklemek, yapmak. Seymour, 1959'da Cambridge Üniversitesi'nde matematik bilim dalında doktorasını aldıktan sonra Cenevre'ye taşındı ve İsviçreli psikolog Jean Piaget'le çalışmaya başladı. Binlerce çocukla yapılan dikkatli gözlem ve röportaj sayesinde Piaget, çocukların dünyanın her yerindeki insanlar ve nesnelere kurduğu günlük etkileşimler yoluyla bilgiyi aktif olarak inşa ettiğini keşfetti. Bilgi çocuklara, bir vazoya su döker gibi verilemez. Aksine, çocuklar oyuncakları ve arkadaşlarıyla oynarken dünya hakkındaki kendi teorilerini sürekli yaratır, revize eder ve test eder. Çocuklar, Piaget'in yapılandırıcı öğrenme teorisine göre, bilgiyi aktif olarak inşa ederler, pasif olarak almazlar. Çocuklar fikir almaz, aksine fikirleri üretirler.

1960'ların başında Seymour, İsviçre Cenevre'den tekrar Cambridge, Massachusetts'e taşındı ve MIT'de öğretim üyesi pozisyonuna getirildi. Bu esnada Seymour çocuk gelişimi konusundaki devrimin merkez üssünden bilgisayar teknolojisinde devrimin merkez üssüne geçiyordu ve sonraki on yıllar boyunca vaktini iki

devrim arasında bağlantı kurarak geçirdi. Seymour MIT'e geldiğinde, bilgisayarlar halen yüz binlerce dolara veya daha fazlasına mal oluyor ve sadece büyük şirketler, devlet kurumları ve üniversitelerde kullanılıyordu. Ancak Seymour, nihayetinde bilgisayarların herkes için, hatta çocuklar için bile erişilebilir olacağını öngördü ve bilgisayarların, çocukların öğrenme ve oynama biçimlerini nasıl değiştirebileceğine dair bir vizyon geliştirdi.

Seymour kısa süre içinde, bilgisayarların eğitime nasıl entegre edileceği konusunda güçlü bir entelektüel mücadelede lider olarak ortaya çıktı. Birçok araştırmacı ve eğitimcinin benimsemeye başladığı bilgisayar destekli eğitim metodunda, bilgisayarlar öğretmen rolüne bürünmektedir: Öğrenciye bilgi ve yönergeyi sunar, öğrencilerin neler öğrendiklerini ölçmek için sınavlar yapar, ardından öğrencinin sınav sorularına verdiği yanıtlara göre sonraki dersleri uyarlar.

Seymour'un tamamen farklı bir vizyonu vardı. Seymour'a göre bilgisayarlar öğretmenin yerini alamazdı, yeni bir ifade aracı olabilirdi; bir şeyler üretmeye yarayacak bir araç olabilirdi. 1971'de, ilk kişisel bilgisayarın kullanıma sunulmasından beş yıl önce, Seymour, Cynthia Solomon'la birlikte "Bir Bilgisayara Yapılabilecek Yirmi Şey" başlıklı bir makalenin eş yazarlığını yaptı. Makale, çocukların resim çizmek, oyun oluşturmak, robotları kontrol etmek, müzik bestelemek ve diğer birçok yaratıcı etkinliği yapmak için bilgisayarları nasıl kullanabileceğini anlatıyordu.

Seymour'un yaklaşımı, Piaget'ten öğrendiklerinin üzerine ekleyerek, çocukların bilgiyi aktif olarak inşa ettiğini, pasif olarak almadığını savunuyordu. Seymour bir adım daha ileri gitti ve çocukların dünyada nesnelere inşa etmede aktif olarak rol aldıklarında, yani bir şeylerin üreticileri olduklarında, bilgiyi en verimli şekilde yapılandırdıklarını savundu. Seymour bu yaklaşımına konstrüktivizm adını verdi, çünkü iki tip yapıyı bir ara-

ya getiriyordu: Çocuklar dünyada bir şeyler inşa ederken, kendi kafalarında-kendilerini bu dünyada ve sonsuza dek bitmeyen bir öğrenme sarmalında yeni şeyler inşa etmeye teşvik eden yeni fikirler inşa ederler.

Seymour ve arkadaşları, bu fikirleri hayata geçirmek amacıyla çocuklar için Logo adını verdikleri yeni bir bilgisayar programlama dili geliştirdiler. O zamana kadar programlama, yalnızca ileri seviye matematik altyapısına sahip insanlarca erişilebilen özel bir faaliyet olarak görülüyordu. Ancak Seymour, programlamayı bilgisayarda bir şeyler üretmek için gerekli evrensel bir dil olarak görüyor ve herkesin programlamayı öğrenmesi gerektiğini savunuyordu.

Kitabı *Mindstorms*'ta, Seymour “Bilgisayar çocuğu programlamak için kullanılır” fikrini savunan bilgisayar destekli eğitim yaklaşımına “çocuk bilgisayarı programlar” diyen kendi yaklaşımıyla karşı çıktı. Program öğrenme sürecinde çocuk, hem “en modern ve en güçlü teknolojinin bir parçası üzerine hâkimiyet kurabileceğini hisseder” hem de “fen, bilim, matematik ve entelektüel model kurma sanatları ile arasında en derin düşüncelerle samimi bir ilişki kurar.”

Logo ilk geliştirildiğinde, çocuklar öncelikle bir robot “kaplumbağanın” hareketlerini kontrol etmek için kullanıyorlardı (elektronik aksamını korumak için yarım küre şeklinde bir kabuğu olduğundan bu isim verilmişti). 1970'lerin sonunda kişisel bilgisayarlar piyasaya ilk çıktığında, çocuklar Logo'yu ekrana resimler çizmek, “ekran kaplumbağasına” nasıl hareket edeceğini, döneceğini ve hareket edeceğini söylemek için “ileri 100” ve “sağa 60” gibi komutlar yazmak için kullandılar. Çocuklar Logo'yla program yazarken, ilgilendikleri projeler üzerinde çalışmak bağlamında matematiksel fikirleri anlamlı ve motive edici bir şekilde öğrendiler.

1980'ler boyunca binlerce okul milyonlarca öğrenciye Logo'da nasıl program yapacağını öğretti, ancak başlangıçtaki coşku sürmedi. Birçok öğretmen ve öğrenci Logo'da program yapmakta zorlandı, çünkü program dili sezgisel olarak öğrenilemeyecek söz dizimi ve noktalama işaretleriyle doluydu. Daha da kötüsü, Logo genellikle öğretmenlerin veya öğrencilerin ilgisini çekmeyen et-kinlikler yoluyla tanıtıldı. Birçok sınıfta Logo öğrencilerin kendi-ni ifade etmesinin ve Seymour'un "güçlü fikirler" dediği şeyleri keşfetmenin bir yolu olarak değil, en son seçim olarak öğretildi. Kısa bir süre önce çoğu okul bilgisayarların diğer kullanımlarına yöneldi. Bilgisayarları, Seymour'un hayal ettiği gibi üretmek ve yaratmak için değil, bilgi aktarımı ve erişimi için araçlar olarak görmeye başladılar.

Seymour'un üreterek öğrenmeye dair fikirleri, Maker Hareketi'nin yükselişi ile kanıtlandığı üzere bir kez daha ilgi çekmeye başladı. Seymour'un Logo konusundaki çalışmaları 50 yılı aşkın bir süre önce başlamış olsa da, çığır açan kitabı *Mindstorms* 1980 yılında basılmış olsa da savunduğu temel fikirler bugün de eskiden olduğu kadar önemlidir ve günümüzde de geçerlidir.

• • •

AKICILIK

Son birkaç yılda, bilgisayar programlama ya da bugünlerde yaygın olarak kullanılan adıyla kodlama öğrenmeye olan ilgi arttı. Artık çocukların kod öğrenmeye başlamasına yardımcı olacak binlerce uygulama, internet sitesi ve atölye var. Scratch prog-

ramlama yazılımı da bu eğilimin bir parçasıdır, ancak belirgin bir fark bulunur.

Kodlamaya giriş genel olarak bulmacaları temel alır. Çocuklardan bir hedefe ulaşmak adına engellerin üzerinden geçecek sanal karakter için bir program hazırlamaları istenir. Örneğin *Star Wars* robotu BB-8'i düşmanla karşılaşmadan hurda metalleri toplayacak şekilde hareket ettir veya R2-D2'yi isyancı pilotlara mesaj götürmek üzere programla. Çocuklar, bu bulmacaları çözmek için programlar oluştururken temel kodlama becerilerini ve bilişim kavramlarını öğrenirler.

Scratch ile bulmacalar yerine projelere odaklanmayı tercih ederiz. Çocuklara Scratch'i tanıtarak onları kendi interaktif hikâyelerini, oyunlarını ve animasyonlarını yaratmaya teşvik ederiz. Fikirlerle başlayıp bunları diğer insanlarla paylaşabilecekleri projelere dönüştürürler.

Neden projelere odaklanmak? Kodlamayı akıcılık ve ifadenin bir biçimi olarak görüyoruz, tıpkı yazı yazmak gibi. Yazmayı öğrendiğinizde, imla, dil bilgisi ve noktalama işaretlerini öğrenmek yeterli olmaz. Hikâye anlatmayı ve fikirlerinizi aktarmayı öğrenmeniz de önemlidir. Aynı şey kodlama için de geçerlidir. Bulmacalar, kodlamanın temel dil bilgisi ve noktalama işaretlerini öğrenmeniz için iyi olabilir, ancak kendinizi ifade etmeyi öğrenmenize yetmez. Sadece kare bulmacalar üzerinde çalışarak yazı yazmayı öğrenmeye çalıştığınızı düşünün. Bu sizin imla ve sözcük dağarcığınızı geliştirebilir, hatta eğlenceli bile olabilirdi, ancak hikâyeler anlatabilecek ve fikirlerinizi akıcı biçimde ifade edebilecek iyi bir yazar olur muydunuz? Hiç sanmıyorum. Proje-ye dayalı bir yaklaşım, ister yazma için ister kodlama için olsun, akıcılık için en iyi yoldur.

Çoğu kişi profesyonel gazeteci ya da roman yazarı olmak için doğmasa da herkesin yazmayı öğrenmesi önemlidir. Aynı şey

benzer nedenlerden ötürü kodlama için de geçerlidir. Çoğu kişi, profesyonel programcı veya bilgisayar mühendisi olmak için de doğmaz, ancak akıcı bir şekilde kodlamayı öğrenmek herkes için değerlidir. İster yazma ister kodlama alanında olsun, akıcı olmak, *düşüncenizi geliştirmenize, sesinizi geliştirmenize ve kimliğinizi geliştirmenize yardımcı olur.*

DÜŞÜNME BİÇİMİNİZİ GELİŞTİRMEK

Yazma sürecinde, fikirlerinizi organize etmeyi, rafine etmeyi ve düşüncelerinizi yansıtmayı öğrenirsiniz. Daha iyi bir yazar olduğunuzda, daha iyi bir düşünür haline gelirsiniz.

Kodlamayı öğrenirken de daha iyi bir düşünür olursunuz. Örneğin karmaşık problemleri daha basit parçalara nasıl ayıracağınızı öğrenirsiniz. Sorunları nasıl tespit edeceğinizi ve çözeceğinizi öğrenirsiniz. Zamanla tasarımlarınızı tekrarlarla iyileştirmeyi ve geliştirmeyi öğrenirsiniz.

Bilgisayar mühendisi Jeannette Wing'in popüler hale getirdiği sayısal düşünme kavramı bu tip stratejilere atıfta bulunur.

Sayısal düşünme stratejilerini öğrenmek, sadece kodlama ve bilişimde değil, problem çözme ve tasarım faaliyetlerinin her türünde size yararlı olabilir. Bilgisayar programlarındaki hatayı düzeltmeyi öğrenerek mutfakta bir tarif işe yaramadığında nerede hata yaptığınızı veya birilerinin talimatlarını izleyerek kaybolduğunuzda neyin yanlış gittiğini anlamanız gereken durumlara daha hazırlıklı olursunuz.

Bulmaca çözmek, bu sayısal-düşünme becerilerinin bazılarını geliştirmenizde yardımcı olabilir, ancak kendi projelerinizi oluşturmak sizi daha ileri götürür ve ifade biçiminizi, hatta kimliğinizi geliştirmenize yardımcı olur.

İFADE BİÇİMİNİZİ GELİŞTİRMEK

Yazı yazma ve kodlama: Her ikisi de ifade biçimi olmasının yanı sıra, fikirleriniz aracılığıyla başkalarıyla iletişim kurmanın da yoludur. Örneğin yazı yazmayı öğrendiğinizde bir arkadaşınıza doğum günü mesajı gönderebilir, yerel gazetenin köşelerine yazı gönderebilir ya da duygularınızı bir günlüğe kaydedebilirsiniz.

Kodlamayı yazı yazmanın bir uzantısı olarak görüyorum; çünkü size “interaktif hikâyeler, oyunlar, animasyonlar ve simülasyonlar” gibi yeni türlerde “yazma” imkânı sağlar. Scratch çevrimiçi topluluğundan bir örnek vereyim. Birkaç yıl önce, Anneler Günü’nden bir önceki gün, Scratch’i annem için interaktif bir Anneler Günü kartı yapmak üzere kullanmaya karar verdim. Başlamadan önce, başkasının Scratch ile Anneler Günü kartı hazırlayıp hazırlamadığını kontrol ettim. Arama kutusuna “Anneler Günü” yazdım ve düzinelerce projeyi görmekten büyük mutluluk duydum, birçoğu son 24 saat içinde benim gibi son dakikacılar tarafından üretilmişti!

Örneğin projelerden biri kocaman kırmızı bir kalp üzerine yazılmış “ANNELER GÜNÜN KUTLU OLSUN” sözleriyle başlıyordu. Harflerin her biri interaktif, fare imleciyle dokunulduğunda bir kelimeye dönüşüyordu. İmleci ekran boyunca her harfe dokunarak taşıdığım özel bir Anneler Günü mesajı ortaya çıkıyordu: “Seni çok seviyorum. Anneler Günü Kutlu Olsun Anneciğim.”

Bu projenin yaratıcısı, Scratch ile ifade biçimini açık bir şekilde geliştiriyordu; kendini yeni yollarla ifade etmeyi ve kodlamayı gündelik hayatının akışına entegre etmeyi öğreniyordu. Gelecekte, gençlerin kendilerini kodlama ile ifade etmelerinin kendilerini en az yazı yazarak ifade etmeleri kadar doğal bir şey olacağına inanıyorum.

(Bu arada o gün annem için Anneler Günü kartı yapmadım. Bunun yerine ona Scratch internet sitesinde bulduğum bir düzine Anneler Günü projesinin linkini gönderdim. Kendisi de yaşam boyu eğitimci olan annem şu mesajla yanıt verdi: “Mitch, tüm çocukların Scratch kartlarına baktım ve çok beğendim... ve çocuklara bu şekilde kutlama yapacak araçları vermeye yardım eden bir çocuğun annesi olduğum için mutluyum!”

KİMLİĞİNİZİ GELİŞTİRMEK

İnsanlar yazmayı öğrendiklerinde, kendilerini ve toplumdaki rollerini farklı şekilde görmeye başlarlar. Brezilyalı eğitimci felsefeci Paulo Freire, yoksul toplumlardaki okuma yazma kampanyalarına, sadece insanların iş bulmalarına yardımcı olmak için değil, aynı zamanda insanların “kendilerini baştan yaratmayı ve yeniden yapabileceklerini” öğrenmelerine yardımcı olmak için öncülük etmiştir. (Tıpkı *Pedagogy of Indignation [Öfkenin Pedagojisi]* kitabında yazdığı gibi).

Kodlama için de aynı potansiyeli görüyorum. Günümüz toplumunda, dijital teknolojiler olasılık ve ilerlemenin sembolüdür. Çocuklar kendilerini ifade etmek ve fikirlerini kodlama yoluyla paylaşmak için dijital teknolojileri kullanmayı öğrendiklerinde kendilerini yeni bir boyutta görmeye başlarlar. Topluma aktif olarak katkıda bulunma ihtimaline inanmaya başlarlar. Kendilerini geleceğin bir parçası olarak görmeye başlarlar.

Scratch’i gençlere tanıtırken, yarattıkları şeyden ve bu süreçte öğrendiklerinden dolayı heyecanlanıyorum. Ancak beni en çok heyecandıran şey, birçok Scratch kullanıcısının kendisini yaratıcı olarak görmeye başlaması ile bir şeyler yaratma ve yeni teknolojilerle kendisini akıcı bir şekilde ifade etme yeteneğine duydukları güven ve gururun gelişmesi.